



Universidad Carlos III de Madrid

Ingeniería en Informática

Proyecto Fin de Carrera

Desarrollo de una aplicación web para la
gestión de información de evaluación

Autora: Paloma Fernández Ceballos

Tutora: Carmen Luisa Padrón Nápoles

A mis padres Ángel y Blanca, mis hermanos Ángel y José Carlos, a mis abuelos y demás familia; a mis pequitas Elena, Patri y Blanca; a los “The Ones”: Adrián, Álvaro, Ana, Conchi e Iván y a todo aquel que en algún momento de todos estos años me ha mostrado su apoyo y ha creído en mi.

Simplemente, GRACIAS.

Paloma



Índice del documento

Índice del documento.....	2
Índice de figuras	4
Índice de tablas	7
1. INTRODUCCIÓN	8
1.1. Objetivos	10
1.2. Planificación y método	12
1.3. Estructura del documento	13
2. PLANTEAMIENTO DEL PROBLEMA	15
2.1. Usabilidad en Interacción Persona-Ordenador	15
2.1.1. Usabilidad de sistemas interactivos	16
2.1.2. Evaluación de la usabilidad	17
2.2. Los cuestionarios en la evaluación de la usabilidad.....	20
2.2.1. Cuestionarios de usabilidad existentes.....	22
2.2.2. Herramientas para la creación de cuestionarios	26
3. SOLUCIÓN DEL PROBLEMA	29
3.1. Arquitectura de la solución	29
3.2. Tecnologías utilizadas	30
3.2.1. Java	30
3.2.2. Struts	36
3.2.3. Hibernate	38
3.2.4. MySQL y JDBC	39
3.2.5. JUnit	40
3.3. Entorno de desarrollo	42
4. ANÁLISIS DE LA APLICACIÓN.....	44
4.1. Especificación de los requisitos de usuario	44
4.1.1. Requisitos de capacidad	44
4.1.2. Requisitos de restricción.....	46
4.2. Casos de uso.....	47
4.2.1. Descripción de Actores.....	47
4.2.2. Descripción de Casos de Uso.....	48
4.2.3. Diagrama de Casos de Uso	77
4.3. Modelo conceptual	80
5. DISEÑO DE LA APLICACIÓN.....	81
5.1. Requisitos de software	81
5.1.1. Requisitos funcionales.....	81
5.1.2. Requisitos de interfaz	86
5.1.3. Requisitos operacionales	87
5.1.4. Requisitos de portabilidad.....	88
5.1.5. Requisitos de documentación	88
5.2. Diseño de la capa de persistencia	89
5.3. Diseño de la capa de lógica de negocio.....	92
5.4. Diseño de la interfaz gráfica	101
6. MANUAL DE USUARIO	103
6.1. Manual de configuración de la aplicación	103
6.1.1. Diagrama de despliegue	103
6.1.2. Pasos para la instalación y configuración de la aplicación	104
6.2. Acceso a la aplicación y menú principal	111
6.3. Tipos de usuarios.....	112



6.4. Operaciones básicas	113
6.4.1. Loguearse	113
6.4.2. Cerrar sesión	113
6.4.3. Modificar datos de usuario	114
6.4.4. Responder un cuestionario.....	115
6.5. Operaciones exclusivas para el administrador	117
6.5.1. Gestión de usuarios	117
6.5.2. Gestión de cuestionarios	120
6.5.3. Gestión de secciones	126
6.5.4. Gestión de preguntas.....	131
6.5.5. Estadísticas	134
7. GESTIÓN DEL PROYECTO	138
7.1. Planificación del proyecto	138
7.2. Análisis de costes.....	143
7.2.1. Personal.....	143
7.2.2. Material e infraestructuras	144
7.2.3. Otros.....	145
7.2.4. Resumen de costes.....	146
8. CONCLUSIONES	147
8.1. Conclusiones finales	147
8.2. Trabajo futuro y ampliaciones.....	149
REFERENCIAS	150



Índice de figuras

Figura 1. Arquitectura de la solución	29
Figura 2. Patrón MVC	30
Figura 3. Clasificación de la Plataforma Java.....	32
Figura 4. Funcionamiento de la especificación JSP 1.2.....	33
Figura 5. Funcionamiento de un Servlet genérico	34
Figura 6. Contenedor de EJB	35
Figura 7. Servidor de aplicaciones.....	37
Figura 8. Arquitectura Hibernate.....	39
Figura 9. Drivers de JDBC	40
Figura 10. CU01- Loguearse.....	48
Figura 11. CU02- Cerrar sesión.....	49
Figura 12. CU03- Dar de alta a usuario.....	50
Figura 13. CU04- Dar de baja a usuario	51
Figura 14. CU05- Modificar datos de usuario.....	53
Figura 15. CU06- Consultar datos de usuario.....	54
Figura 16. CU07- Crear cuestionario.....	55
Figura 17. CU08- Eliminar cuestionario	57
Figura 18. CU09- Modificar cuestionario	59
Figura 19. CU10- Consultar cuestionario	60
Figura 20. CU11- Crear sección	61
Figura 21. CU12- Eliminar sección.....	63
Figura 22. CU13- Modificar sección	65
Figura 23. CU14- Consultar sección.....	66
Figura 24. CU15- Crear pregunta	67
Figura 25. CU16- Eliminar pregunta	68
Figura 26. CU17- Modificar pregunta	70
Figura 27. CU18- Consultar pregunta	71
Figura 28. CU19- Completar cuestionario	73
Figura 29. CU20- Mostrar promedios de puntuaciones por pregunta y característica	75
Figura 30. CU21- Mostrar porcentaje de usuarios que han respondido una de opción de cada pregunta y característica	77
Figura 31. Escenario 0: Login de usuario.....	78
Figura 32. Escenario 1: Funcionalidades usuario básico.....	78
Figura 33. Escenario 2: Funcionalidades usuario administrador	79
Figura 34. Diagrama de clases de análisis	80
Figura 35. Diagrama relacional de la base de datos	91
Figura 36. Diagrama de clases detallado.....	95
Figura 37. Diagrama de secuencia- Usuario básico	96
Figura 38. Diagrama de secuencia- Gestión de usuarios	97
Figura 39. Diagrama de secuencia- Gestión de cuestionarios	98
Figura 40. Diagrama de secuencia- Gestión de secciones	99
Figura 41. Diagrama de secuencia- Gestión de preguntas.....	100
Figura 42. Diagrama de secuencia- Visualización de estadísticas	101
Figura 43. Diseño de la interfaz gráfica.....	102
Figura 44. Módulos utilizados arquitectura J2EE	104
Figura 45. Inserción script base de datos	105
Figura 46. Conexión a la base de datos.....	109



Figura 47. Estructura de carpetas de JBoss	109
Figura 48. Iniciar JBoss	110
Figura 49. JBoss iniciado	110
Figura 50. Configuración variables de entorno del sistema.....	111
Figura 51. Menú administrador	112
Figura 52. Menú usuario básico	112
Figura 53. Página para loguearse	113
Figura 54. Página para cerrar sesión	113
Figura 55. Página que muestra como elegir una opción del menú	114
Figura 56. Página para modificar los datos de un usuario	114
Figura 57. Página para modificar la contraseña de usuario.....	115
Figura 58. Página para elegir la opción de responder un cuestionario	115
Figura 59. Página para responder una pregunta del cuestionario	116
Figura 60. Página que muestra que el cuestionario respondido con éxito	116
Figura 61. Página para dar de alta a usuario.....	118
Figura 62. Página que muestra que un usuario ha sido dado de alta con éxito.....	118
Figura 63. Página para consultar datos de un usuario	119
Figura 64. Página para elegir la opción de dar de baja a un usuario	120
Figura 65. Página para dar de baja a un usuario	120
Figura 66. Página para crear un cuestionario	121
Figura 67. Página para consultar un cuestionario.....	122
Figura 68. Página que muestra los datos del cuestionario consultado	122
Figura 69. Página que muestra las preguntas de una sección perteneciente a un cuestionario	123
Figura 70. Página para modificar un cuestionario	124
Figura 71. Página para añadir secciones a un cuestionario.....	124
Figura 72. Página para añadir secciones (previamente creadas) a un cuestionario.....	125
Figura 73. Página que muestra que un cuestionario ha sido modificado con éxito.....	125
Figura 74. Página para eliminar un cuestionario	126
Figura 75. Página para crear una sección	127
Figura 76. Página para consultar los datos de una sección.....	128
Figura 77. Página para elegir la sección a modificar.....	129
Figura 78. Página para modificar los datos de una sección.....	129
Figura 79. Página para añadir una pregunta nueva a la sección seleccionada	130
Figura 80. Página para eliminar una sección	130
Figura 81. Página para crear una pregunta.....	131
Figura 82. Página para consultar los datos de una pregunta.....	132
Figura 83. Página para modificar los datos de una pregunta	133
Figura 84. Página que muestra que la pregunta se ha modificado con éxito	133
Figura 85. Página para eliminar una pregunta.....	134
Figura 86. Página para seleccionar el cuestionario del que se desean ver sus promedios	135
Figura 87. Página que muestra las preguntas del cuestionario seleccionado	135
Figura 88. Página que muestra las puntuaciones medias por característica y preguntas.....	136



Figura 89. Página que indica que el cuestionario ha llegado a su fin	136
Figura 90. Página que muestra el porcentaje de usuarios que ha respondido cada opción	137
Figura 91. Ciclo de vida en cascada.....	138
Figura 92. Planificación de tareas del proyecto.....	139
Figura 93. Diagrama de Gantt	142



Índice de tablas

Tabla 1. RC1- Gestión de usuarios	44
Tabla 2. RC2- Gestión de cuestionarios.....	44
Tabla 3. RC3- Gestión de secciones.....	45
Tabla 4. RC4 Gestión de preguntas	45
Tabla 5. RC5- Visualizar estadísticas	45
Tabla 6. RC6- Completar cuestionario.....	46
Tabla 7. RR1- Seguridad de la aplicación	46
Tabla 8. RR2- Fácil manejo	46
Tabla 9. RR3- Lenguaje en castellano	47
Tabla 10. RF01- Dar de alta a usuario	81
Tabla 11. RF03- Consultar datos personales de usuario.....	81
Tabla 12. RF02- Dar de baja a usuario.....	81
Tabla 13. RF04- Modificar datos personales de un usuario	82
Tabla 14. RF05- Crear cuestionario	82
Tabla 15. RF06- Eliminar cuestionario.....	82
Tabla 16. RF07- Consultar cuestionario.....	82
Tabla 17. RF08- Modificar cuestionario	83
Tabla 18. RF09- Crear sección.....	83
Tabla 19. RF10- Eliminar sección	83
Tabla 20. RF11- Consultar sección	83
Tabla 21. RF12- Modificar sección	84
Tabla 22. RF13- Crear pregunta	84
Tabla 23. RF14- Eliminar pregunta.....	84
Tabla 24. RF15- Consultar pregunta.....	84
Tabla 25. RF16- Modificar pregunta.....	85
Tabla 26. RF17- Completar cuestionario.....	85
Tabla 27. RF18- Visualizar promedios de puntuación por pregunta y característica	85
Tabla 28. RF19- Visualizar porcentaje de usuarios que han respondido cada opción de una pregunta y característica	85
Tabla 29. RF20- Acceso a la aplicación mediante login de usuario y contraseña	86
Tabla 30. RF21- Cerrar sesión	86
Tabla 31. RNF01- Interfaz intuitiva y de fácil manejo.....	86
Tabla 32. RNF02- Lenguaje de la interfaz: Castellano	87
Tabla 33. RNF03- Interfaz optimizada para IE y Firefox.....	87
Tabla 34. RNF04- Las consultas a la base de datos no excederán los tres segundos	87
Tabla 35. RNF05- Aplicación Web J2EE + Struts + Hibernate	88
Tabla 36. RNF06- Sistema gestor de base de datos MySQL 5.0	88
Tabla 37. RNF07- Servidor de aplicaciones JBoss 1.4.x	88
Tabla 38. RNF08- Entrega de documentación adjunta	88
Tabla 39. Script de la base de datos.....	108
Tabla 40. Costes del personal implicado en el proyecto	144
Tabla 41. Resumen de días y costes totales del personal	144
Tabla 42. Costes de materiales e infraestructuras	145
Tabla 43. Otros gastos	145
Tabla 44. Resumen de costes del proyecto	146



1. INTRODUCCIÓN

En el ámbito de la Interacción Persona-Ordenador tiene especial relevancia el campo dedicado a estudiar la usabilidad de las aplicaciones. La usabilidad suele definirse como un atributo que, dentro del conjunto general de los que conforman un sistema de información, permite que los usuarios de dichos sistemas los utilicen más rápida y fácilmente (Dumas & Redish 2000), lo que redundaría en la eficiencia en la realización de la tarea y en una mayor satisfacción del usuario que la lleva a cabo (ISO 9241 1998), influyendo directamente ambos factores en la productividad.

Durante las fases de diseño y desarrollo de sistemas de información, las decisiones que se lleven a cabo afectan directa o indirectamente a la usabilidad de las aplicaciones. Para ello, es necesario aplicar procesos de Ingeniería de la Usabilidad en la construcción de las aplicaciones.

La técnica más habitual utilizada para medir la usabilidad desde el punto de vista del usuario, son los cuestionarios. En el ámbito de la usabilidad, los cuestionarios forman parte de técnicas empíricas indirectas, ya que no estudian la interfaz en sí misma, sino que lo hacen a través de las opiniones de los usuarios. Un cuestionario es una prueba psicométrica de actitud que mide determinados factores para los cuales se diseña, y que conlleva dos medidas fundamentales que aseguran la calidad del mismo y la ausencia de sesgos: la fiabilidad (Muñiz 1996) y la validez (Paz 1996), no obstante, en el campo de la usabilidad se justifica el uso de cuestionarios de los cuales no se haya comprobado el cumplimiento de estos requisitos, en la mayoría de los casos con el objetivo de poder realizar una evaluación más económica.

La evaluación de la usabilidad basada en cuestionarios es especialmente útil para estudiar la satisfacción subjetiva de los usuarios, así como para estudiar la forma en la que los usuarios utilizan la aplicación qué características de la misma les gustan más o menos. Además, como los cuestionarios no necesitan supervisión para ser realizados, llegan a una gran cantidad de población, permitiendo descubrir importantes diferencias y/o necesidades entre grupos de usuarios distintos.

Actualmente existen a disposición de los desarrolladores cuestionarios genéricos probados para evaluar la usabilidad como *QUIS* (Chin et al. 1988), *IsometricsL* e *IsometricsS* (Gediga et al. 1999), *PUTQ* (Lin et al. 1997), *CSUQ* (Lewis 1995), *PUEU* (Davis 1989) o *WAMMI* (Kirakowski & Cierlik 1998), sin embargo, estos cuestionarios a veces no son directamente aplicables dependiendo de la naturaleza del sistema a evaluar, lo que hace necesaria la definición de nuevos cuestionarios más específicos.



Cabe tener en cuenta que tanto si se utilizan cuestionarios previamente definidos como si se desarrollan íntegramente, se han detectado inconvenientes en la aplicación de cuestionarios en procesos de Ingeniería de Usabilidad (Mayhew 1999):

- Es necesario calcular las medidas de validez y fiabilidad del cuestionario, consiguiendo que ésta última mantenga un buen compromiso con la longitud del mismo.
- Se debe seleccionar previamente los sujetos que van a realizar el cuestionario de acuerdo al perfil o perfiles de usuarios que se quiera evaluar, esto será para conseguir un cuestionario correctamente distribuido.
- Los cuestionarios obtienen sus resultados a través de medidas de distintos factores entre los que no se puede establecer ninguna relación, por lo que es importante tener en cuenta que del análisis de los datos obtenidos no se puede obtener de manera precisa los elementos de la interfaz evaluada (o ausencia de los mismos) que dificultan al usuario la interacción con el sistema, ni se pueden comparar directamente los resultados obtenidos con diferentes cuestionarios.

Estos problemas podrían solventarse si se tuviera un cuestionario previamente definido para un objetivo concreto, pero habría que tener en cuenta los costes que supondrían la adquisición del mismo y el posterior análisis de los datos recogidos, ya que la recolección de datos suele ser realizada por la entidad externa que definió el cuestionario.

A continuación se resumen algunas técnicas de aprendizaje automático que podrían paliar algunos de los inconvenientes mencionados (García, Sicilia, González & Hilera 2002) y (García et al., 2001):

- ✚ Para obtener un compromiso adecuado entre la longitud y fiabilidad del cuestionario se tendrían que aplicar reglas de asociación encargadas de buscar ítems redundantes que pudieran ser eliminados. La detección automática de estas redundancias implicaría reducción de esfuerzo en la fase de desarrollo del cuestionario.
- ✚ Una técnica que disminuiría los costes asociados a la selección de las muestras de usuarios que van a responder el cuestionario, sería la generación automática de agrupaciones de evaluadores que realizaría



una selección apropiada y sin sesgos para llevar a cabo el proceso de evaluación.

Para poder aplicar las técnicas arriba mencionadas es imprescindible diseñar un modelo que permita representar cualquier tipo de cuestionarios que intervenga en una evaluación, así como en el entorno que se aplica.

1.1. Objetivos

El objetivo de este proyecto es el diseño y la implementación de una aplicación web capaz de gestionar información de evaluación. Esta información vendrá definida por cuestionarios que irán dirigidos principalmente a la evaluación de la usabilidad de sistemas informáticos.

El presente proyecto fija su atención en la IPO [4] (*Interacción Persona-Ordenador*). A mediados de los 80 se acuña el término HCI (*Human-Computer Interaction*) que se define como "Un lenguaje de entrada para el usuario, de salida para el sistema y un protocolo de interacción" (1981). En 1987, se da una definición más precisa: "HCI es una disciplina relacionada con el diseño, evaluación e implementación de sistemas interactivos para el uso humano y con el estudio de los grandes fenómenos y alrededor de ellos" (ACM SIGCHI, 1987). Aclarar que HCI e IPO vienen a ser lo mismo.

La IPO surge de la necesidad de resolver el conflicto de conciliación entre los seres humanos y la tecnología, imprescindible hoy en día en nuestra vida cotidiana ya que muchas de nuestras herramientas actuales son ordenadores de múltiples diseños con los cuales interaccionamos. Escrito desde una perspectiva multidisciplinar, la IPO presenta los conceptos y la metodología para el diseño de sistemas interactivos centrados en el usuario que permitan que cualquier persona perciba el proceso interactivo como una experiencia efectiva, eficiente y satisfactoria.

Entre los objetivos principales de la IPO se encuentran desarrollar o mejorar la seguridad, utilidad, efectividad, eficiencia y la usabilidad de sistemas que incluyen ordenadores.

Uno de los objetivos principales de este proyecto es contribuir a aumentar la usabilidad de los sistemas informáticos.

La usabilidad es la cualidad de un sistema respecto a su facilidad de uso: múltiples formas de intercambiar información entre el usuario y el sistema; su facilidad de aprendizaje para nuevos usuarios que garantizan interacción efectiva y máximas prestaciones; y la satisfacción del usuario



incluyendo el soporte al usuario para garantizar las metas (robustez) (Rosson y Carroll, 2002).

Algunos de los factores que contribuyen a aumentar la usabilidad de un sistema (Preece 2002) son:

- ✚ Que sea efectivo
- ✚ Que sea eficiente
- ✚ Que sea seguro
- ✚ Que sea útil
- ✚ Que se pueda aprender fácilmente
- ✚ Que sea fácil de recordar cómo se usa

Para que un sistema cumpla estos objetivos, debe satisfacer una serie de requisitos. A su vez estos requisitos se extraerán a raíz de una fase previa de recogida de datos. El objetivo de esta fase es recoger datos suficientes, relevantes y apropiados para definir un conjunto estable de requisitos, en caso de que ya exista este conjunto, servirá para expandir, clarificar y confirmar dicho conjunto.

En esta recogida se deben conocer cómo se realizan las tareas en la actualidad, así como las metas asociadas, el contexto en el que se realizan y las razones de por qué las cosas son como son. Existen varias técnicas para recoger información las cuales no son excluyentes: cuestionarios, entrevistas, grupos de interés y talleres, observación, estudio de documentación y software de registro entre otras.

Los cuestionarios constituyen una de las técnicas más utilizadas para evaluar la usabilidad, y permiten medir directamente la satisfacción e indirectamente otros factores de usabilidad. Sin embargo, el diseño de cuestionarios de usabilidad con las características apropiadas es un proceso costoso y que requiere cierta experiencia en el área, y la falta de medios comunes para diseminar los resultados limitan las posibilidades de análisis y meta-análisis de las evaluaciones.

El objetivo principal de este proyecto es realizar una aplicación web que permita recopilar y procesar toda la información relacionada con un cuestionario, que seguirá un modelo general, con el fin de poder aplicarlo para la evaluación de la usabilidad de cualquier sistema.

El modelo de cuestionario en el que se ha basado la aplicación permite representar cualquier tipo de cuestionarios que intervenga en una evaluación así como en el entorno en el que se aplicarán. Se ha implementado este modelo con el fin de poder tener cuestionarios lo más completos posibles y subsanar en parte los inconvenientes que pudieran tener otros cuestionarios de evaluación más específicos.



1.2. Planificación y método

En esta sección se presentan las fases en las que se ha dividido el desarrollo del proyecto:

En la primera fase se proporciona el estado de la cuestión en lo relativo a la evaluación de la usabilidad, focalizando la atención en el papel (utilización y generación) de los cuestionarios en dicha evaluación.

Esta **primera fase** del proyecto está relacionada con la familiarización del lector con las tecnologías empleadas para el desarrollo del proyecto, así como del contexto para el que se desarrolla la solución propuesta en el proyecto. Además se iniciará al lector en la toma de contacto con aspectos tecnológicos relacionados con Struts [9], MySQL [8], Hibernate [3] y AJAX entre otros.

La **segunda fase** se ha dedicado al Análisis y Diseño de la aplicación, la cual se ha llevado a cabo mediante la obtención de requisitos de usuario, a partir de éstos, se han obtenido los casos de uso y un primer diagrama de clases.

En cuanto a la fase de diseño, se han extraído los requisitos de software, el diagrama relacional de la base de datos obtenida y el diagrama de clases detallado.

La **tercera fase** del proyecto ha consistido en la preparación de la infraestructura de hardware y software de la aplicación. A partir de lo aprendido en las fases anteriores, se ha instalado y configurado el entorno de desarrollo, en este caso, MyEclipse (es un *plug-in* de Eclipse [2]), las bibliotecas Java necesarias, el servidor de aplicaciones JBoss [6], así como el driver conector con nuestra base de datos MySQL.

Una vez configurado el entorno, se ha procedido a la implementación del código en Java [5]. Para ello, lo primero ha sido el diseño básico de las interfaces de usuario, donde se han plasmado las funcionalidades descritas en los casos de uso.

A continuación se ha pasado a implementar la lógica de negocio. Se ha creado una clase para cada entidad lógica ocupada de la administración de la misma y se ha ido probando cada funcionalidad por separado, mediante clases *Test* de prueba con JUnit [7].

Teniendo las interfaces por un lado, y la lógica de negocio por otro, se ha procedido a integrar la aplicación mediante Struts [9] que ha facilitado la implementación de la arquitectura *Modelo-Vista-Controlador*.



A partir de aquí, se han vuelto a hacer pruebas ya con toda la aplicación completa y una vez comprobado el correcto funcionamiento de la misma, se ha procedido a “dar color” a las interfaces para dejar por fin, una aplicación fácil de usar así como agradable a la vista del usuario más detallista.

Cabe destacar que para asegurar la obtención de buenos resultados, los documentos generados en cada fase han sido revisados y completados a medida que se ha ido desarrollando el proyecto. Estos documentos incluyen información sobre su versión, que se corresponde con el número de la fase de desarrollo concluida.

1.3. Estructura del documento

El contenido del presente documento está estructurado como se describe a continuación:

- 1. Introducción:** En este capítulo se expone brevemente la situación que ha impulsado el desarrollo de este proyecto, explicando así cuál es la finalidad del mismo. Se describe la metodología seguida para su elaboración, así como la estructura que sigue el presente documento.
- 2. Planteamiento del problema:** En este capítulo se ofrece una descripción del estado del arte de las investigaciones y trabajos desarrollados en el área de la evaluación de la usabilidad. Se hace una breve introducción a la usabilidad dentro de la interacción persona-ordenador y se explica el papel que tienen los cuestionarios en la evaluación de ésta.
- 3. Solución del problema:** Este capítulo representa la arquitectura implementada para el desarrollo del presente proyecto, así como las tecnologías y el entorno de desarrollo empleados para llevarlo a cabo.
- 4. Análisis de la aplicación:** En este capítulo se exponen los requisitos de usuario extraídos a través de las reuniones con el cliente, así como los casos de uso derivados de los mismos y un primer diagrama de clases obtenido a partir de los anteriores. Todos ellos servirán de referencia para la fase de diseño.
- 5. Diseño de la aplicación:** En este capítulo se ofrece el diseño de la aplicación. Esta fase la componen los requisitos de software, el



diseño de la base de datos, el diseño detallado de la lógica de negocio así como el diseño de la interfaz.

- 6. Manual de usuario:** Este capítulo contiene el manual de configuración, necesario para poner en funcionamiento la aplicación, así como un breve manual de uso de la misma con el objetivo de mostrar al usuario su funcionamiento.
- 7. Gestión del proyecto:** Ofrece la planificación del proyecto, describiendo las tareas a seguir así como el tiempo planificado para cada una de ellas. Añade también un presupuesto orientativo del coste de la aplicación.
- 8. Conclusiones:** En este capítulo, la autora del proyecto expone sus reflexiones acerca de la consecución de los objetivos marcados y esboza líneas de trabajo futuras para la posible ampliación del proyecto.



2. PLANTEAMIENTO DEL PROBLEMA

Este capítulo así como parte de la introducción del presente documento se extraen de la Tesis Doctoral “Propuesta de un modelo de conocimiento para el uso de cuestionarios en la evaluación de la usabilidad” (García Barriocanal 2004), publicada en la Universidad de Alcalá en abril de 2004 por María Elena García Barriocanal, Ingeniero en Informática.

En él se ofrece una descripción del estado del arte de las investigaciones y trabajos desarrollados en el área de la evaluación de la usabilidad mediante cuestionarios relevantes para los objetivos de este trabajo.

2.1. Usabilidad en Interacción Persona-Ordenador

La Interacción Persona-Ordenador es la disciplina relativa al diseño, evaluación e implementación de sistemas interactivos de computación en lo referente a su uso por los seres humanos, y al resto de aspectos que se centran en torno a dicha relación (Hewett et al. 1996). El ámbito de la Interacción Persona-Ordenador se extiende a su vez a dos tipos de disciplinas: las referidas al ser humano y las referidas a los ordenadores. Para la primera de ellas, los estudios referidos parten de la psicología y sociología, e incluyen teoría de la comunicación, lingüística y psicología cognitiva; Para la segunda, son relevantes las disciplinas referentes al desarrollo de sistemas con los que una persona interactúa para realizar tareas con un ordenador, lo que incluye la computación gráfica, los sistemas operativos, los lenguajes de programación, los entornos de desarrollo y la ingeniería y métodos de desarrollo de sistemas de información, entre otros. La investigación en estos últimos campos, desde el punto de vista de la interacción persona-ordenador, incluye los resultados obtenidos de investigaciones sobre psicología y sociología aplicadas al área, que a su vez dirigen sus esfuerzos a estudiar el comportamiento, reacciones y procesos cognitivos que realizamos los seres humanos en nuestra interacción con el ordenador.

Para conseguir que los usuarios utilicen los sistemas interactivos de manera rápida y fácil, uno de los aspectos más importantes radica en la buena usabilidad de sus interfaces. Por lo que esta área es una de los principales objetos de investigación dentro de la interacción persona-ordenador.



2.1.1. Usabilidad de sistemas interactivos

La usabilidad, según el estándar ISO 9241 "*Ergonomic requirements for work with visual display terminals*" (ISO 9241 1998), se define como la eficiencia, eficacia y satisfacción con la que un grupo concreto de usuarios alcanza unos objetivos específicos utilizando un determinado sistema, entendiendo por eficacia la exactitud y completitud con la que el usuario alcanza sus objetivos, por eficiencia la relación entre los recursos utilizados para la consecución de los objetivos y la exactitud y completitud del resultado, y por satisfacción la comodidad y grado de aceptación del usuario con el sistema de trabajo utilizado. Debido a la poca correlación existente entre ellas, a la hora de medir la usabilidad de un sistema, se deben considerar cada una de estas características de manera independiente (Frokjaer et al. 2000).

Algunos autores consideran la definición anterior demasiado teórica y han buscado indicadores de usabilidad más concretos que faciliten la evaluación y usabilidad de interfaces. A continuación se enumeran (Nielsen 1993):

- ✚ **Facilidad de la interfaz para ser memorizada (*memorability*):** Una interfaz de manejo fácil y sencillo de recordar al usuario aumenta la usabilidad del sistema, permitiendo que el usuario pueda utilizarlo con agilidad incluso después de un periodo sin haber usado la aplicación.
- ✚ **Eficiencia:** El sistema tiene que ser eficiente con la finalidad de que la productividad del usuario aumente.
- ✚ **Facilidad para ser aprendido o facilidad de aprendizaje (*learnability*):** Otra característica importante de los sistemas usables es que el usuario aprende rápida y fácilmente a utilizarlos, de manera que pueda comenzar a trabajar con el sistema lo antes posible.
- ✚ **Ausencia de errores:** La interfaz del sistema debe proporcionar una baja tasa de errores, entendiendo por error cualquier acción que pueda realizar el usuario y que no le conduzca al resultado esperado.
- ✚ **Satisfacción:** Un sistema usable tiene que proporcionar a los usuarios un alto grado de satisfacción subjetiva.

Por otro lado (Dix et al. 1998) propone tres grandes grupos de conceptos que intervienen en la usabilidad de los sistemas al mismo nivel



de abstracción, que son la flexibilidad, la robustez y la facilidad de aprendizaje.

- ✚ **Flexibilidad:** Un sistema será más flexible cuánto mayor cantidad de formas aporte para que el usuario interactúe con él. Para contribuir en el aumento de la flexibilidad del sistema, éste debe permitir la sustitución y migración de tareas, la personalización de la interfaz, y posibilidades variadas de realización de cada tarea.
- ✚ **Robustez:** Una interfaz robusta es aquella que proporciona al usuario un nivel de soporte suficientemente adecuado como para que éste consiga alcanzar sus objetivos. Esto se consigue, por ejemplo, proporcionando buenos tiempos de respuesta a las acciones del usuario.
- ✚ **Facilidad del sistema para ser aprendido:** Un sistema tiene facilidad para ser aprendido si la propia interfaz facilita que el usuario aprenda con rapidez a manejarla. Características de más bajo nivel como la predecibilidad, la sinteticidad, la familiaridad de términos y aspecto, la consistencia y un modo de funcionamiento de la interfaz parecido al de la mayoría de los sistemas, colaboran para conseguir un alto grado de facilidad de aprendizaje.

La medida del grado de usabilidad se puede obtener a través de medidas cualitativas y cuantitativas (Mandel 1997), estas últimas han demostrado ser más valiosas para llevar a cabo diseños de interfaces de usuario usables (Kirakowsky & Cierlic 1998). Algunos ejemplos de medidas cuantitativas (o medidas sobre el proceso de ejecución) son: el número de acciones a realizar por el usuario, el tiempo que necesita para ello, o el número de errores cometidos en la realización de una tarea. Para el caso concreto de páginas Web, otras medidas cuantificables basadas en atributos internos (Brajnik 2000), que caracterizan la usabilidad de las interfaces hacen referencia a la composición de la página (número de palabras, número de enlaces, número de imágenes frente a la cantidad de texto, porcentaje de texto enfatizado y tamaño de la página) y a su formato (número de agrupaciones de texto, número de fuentes y colores distintos utilizados). Las medidas cualitativas o medidas subjetivas se basan siempre en datos orales o escritos sobre la percepción, opinión, preferencias o satisfacción del usuario sobre el sistema.

2.1.2. Evaluación de la usabilidad

Puesto que la usabilidad de las aplicaciones se ve afectada por las diferentes decisiones que puedan adoptarse en fases de diseño y



construcción de sistemas de información, es necesario aplicar procesos de ingeniería de usabilidad (Mayhew 1999) en la construcción de aplicaciones para garantizarla. La ingeniería de usabilidad es una práctica de ingeniería que incluye las teorías, métodos y técnicas para prevenir, encontrar y corregir problemas de usabilidad.

A pesar de que la integración de las técnicas de ingeniería de usabilidad no está totalmente instaurada actualmente en los procesos de desarrollo de aplicaciones, la evaluación de la usabilidad es una práctica extendida. Mediante ella se detectan posibles problemas en la interacción entre el usuario y el sistema, con objeto ya sea de mejorar las interfaces del sistema (evaluación formativa) o bien de elegir la mejor entre dos o más opciones (evaluación sumativa) (Nielsen 1993). La evaluación de interfaces puede definirse como "una valoración de la conformidad entre el funcionamiento real de un sistema y el funcionamiento deseado" (Whitefield et al. 1991), entendiéndose el funcionamiento del sistema desde el punto de vista de las propiedades de la usabilidad que se atribuyen a los mismos.

Existen cuatro formas básicas de llevar a cabo una evaluación de usabilidad (Mack & Nielsen 1994), atendiendo a la forma de llevarla a cabo y no a la técnica de evaluación utilizada, que son:

- ✚ De manera **automática**: obteniendo medidas de usabilidad de las especificaciones de las interfaces a través de programas software.
- ✚ De manera **empírica**: realizando pruebas sobre las interfaces utilizando usuarios reales.
- ✚ De manera **formal**: utilizando modelos exactos y fórmulas para calcular las medidas de usabilidad.
- ✚ De manera **informal**: basándose en "reglas de oro", conocimiento y experiencia de los evaluadores, casi siempre expertos.

En la actualidad, la usabilidad puede evaluarse a través de diferentes métodos, dependiendo del presupuesto y tiempo del que se disponga y de la eficiencia esperada. De hecho, existen muchas variaciones de un mismo método que dependiendo de la clasificación consultada pueden llegar a constituir un tipo por sí solos. Según la técnica de evaluación que se utilice, los métodos pueden clasificarse en las siguientes categorías: de inspección (inspection methods), de investigación o indagación (inquiry methods) y test o prueba de usabilidad (usability testing).

Los métodos de inspección son aquellos en los que un experto en usabilidad examina los aspectos relacionados con la misma en una interfaz de usuario y obtiene una lista con los errores encontrados. Algunos métodos de inspección concretos son la evaluación heurística (Nielsen 1994b), la simulación cognitiva (Wharton et al. 1994), la simulación conjunta (Bias



1994), la inspección de consistencia, de estándares y guías de comprobación (Wixon et al. 1994), y la inspección formal (Kahn & Prail 1994).

Los métodos de investigación o indagación son aquellos en los que se recoge información sobre las opiniones del usuario acerca de su complacencia con la interfaz, de si se comprende su funcionamiento y de si es útil para realizar sus tareas. Los principales tipos de métodos de investigación se pueden clasificar en tres grupos, dependiendo de la manera en la que se produzca la aproximación al usuario para averiguar su opinión: aproximaciones de campo, aproximaciones en grupo y aproximaciones individuales.

En las aproximaciones de campo el evaluador se traslada al entorno donde el usuario trabajará. El método más característico para llevarlas a cabo es la observación de campo (Diaper 1989).

En las aproximaciones de grupo se realizan sesiones fuera del entorno laboral con un grupo de usuarios que haya utilizado el sistema, a los que se pregunta por sus opiniones y preferencias acerca del sistema objeto de evaluación. El método más utilizado para este tipo de aproximación es el de grupos orientados (O'Donnell et al. 1989).

En la aproximación individual la evaluación se lleva a cabo realizando preguntas efectivas a usuarios individuales. Alguno de los métodos que se suele utilizar son los cuestionarios (Root & Draper 1983) y las entrevistas (Guillham 2000) (Foody 1999) (Oppenheim 1992).

Existe además otro tipo de aproximación, no excluyente con las anteriormente expuestas, es la llamada remota o a distancia. Esta aproximación se realiza sin tener contacto directo con el usuario del sistema, utilizando redes telemáticas en la mayoría de los casos. Los métodos más habituales que se utilizan en remoto son los cuestionarios, la evaluación basada en videoconferencias y la captación automatizada de datos.

Con el test de usabilidad se desarrollan experimentos para obtener información específica acerca de un diseño, siempre con usuarios reales (no con expertos), realizando tareas reales y observando y grabando la actuación de los participantes. Para llevar a cabo el test de usabilidad, además de preparar las diferentes fases del mismo, como se describe en la parte II de (Dumas & Redish 2000), hay que aplicar alguna técnica para conducir al usuario a través de las tareas y reunir los datos. Las técnicas son muy variadas, dependiendo de cómo se vaya a guiar al usuario en la sesión. Entre las técnicas más habituales están: el protocolo de pensamiento manifiesto (Nielsen 1993), el método tutorizado (Mack &



Burdett 1992), el método de descubrimiento conjunto o interacción constructiva (O'Malley et al. 1984), el test retrospectivo (Nielsen 1993), o el método de instrucción previa (Vora & Helander 1995).

Como ocurre con algunos métodos de indagación, el test de usabilidad puede llevarse a cabo de manera remota (Hartson 1996), ya sea en distinto lugar y a distinta hora, o en distinto lugar y a la misma hora.

2.2. Los cuestionarios en la evaluación de la usabilidad

Un cuestionario es una herramienta que permite recoger información a través de las preguntas que lo componen, de manera que la información quede grabada para su posterior análisis. En el ámbito de la interacción persona-ordenador, hay tres tipos básicos de cuestionarios, dependiendo de la naturaleza de la información que recojan:

- ✚ **Cuestionarios de recogida de hechos (*factual-type questionnaires*)**: recogen información pública u observable de manera rápida y poco tediosa.
- ✚ **Cuestionarios de opinión (*opinion-type questionnaires*)**: recogen la opinión de la persona que los responde acerca de alguien o algo, sin que existan opciones correctas o incorrectas.
- ✚ **Cuestionarios de actitud o satisfacción (*attitude or satisfaction questionnaires*)**, que recogen el parecer del encuestado en función de experiencias vividas.

Los cuestionarios de satisfacción son los más utilizados en el ámbito de la interacción persona-ordenador, puesto que con ellos se recogen las actitudes de los usuarios respecto a un sistema con el cual han trabajado, siendo herramientas importantes en los distintos procesos de evaluación dentro de la ingeniería de usabilidad.

De acuerdo al ciclo de vida definido en (Mayhew 1999), los cuestionarios se pueden utilizar en todas las fases:

- ❖ En la de análisis de dominio se utilizan cuestionarios de hechos para recoger información sobre los usuarios y componer el perfil de usuario que utilizará el sistema.
- ❖ Durante la de especificación funcional se utilizarán cuestionarios de actitud y de hechos durante el análisis de tareas previo al diseño.



- ❖ Durante el diseño (iterativo) se deben llevar a cabo evaluaciones del prototipo hasta tener la versión definitiva.
- ❖ En el desarrollo se debe preparar el test de usabilidad que se realizará en la siguiente fase.
- ❖ En la fase de pruebas se realizará la evaluación de usabilidad definitiva.

El tipo de cuestionario variará dependiendo del tipo de evaluación que se quiera realizar. En la prueba o test de usabilidad -véase el capítulo 14 de (Dumas & Redish 2000)-, se utilizan cuestionarios de hechos (cuestionarios *pre-test*) en la fase de preparación del test para recoger información sobre los antecedentes de los participantes que permitan hacer la selección adecuada y ayudar a interpretar los resultados finales. Los cuestionarios de actitud (cuestionarios *post-test*) se utilizan una vez que ha finalizado la prueba para recoger información sobre la satisfacción global del participante respecto al sistema. Incluso, en determinadas ocasiones, es interesante conocer la opinión del participante sobre una funcionalidad concreta en el momento justo en el que finalice una tarea o escenario que pruebe dicha parte de la interfaz; en ese caso se utilizaría también un cuestionario de actitud, que se denomina cuestionario post-tarea.

Si se realiza una evaluación heurística, es necesario que se utilicen cuestionarios en la fase posterior a la detección de problemas por parte del experto. En esta fase, denominada de valoración de la severidad del problema, cada experto recibe un cuestionario de opinión en el que valora la importancia de cada problema encontrado, no solo por él, sino por el resto de evaluadores (Nielsen 1994b).

Si la evaluación se lleva a cabo utilizando el cuestionario como método de investigación, su tipo será de actitud, y será respondido por el participante después de haber realizado las tareas correspondientes a cada escenario. Este tipo de cuestionario suele utilizar escalas *Likert* (Foody 1999), en el que todas las preguntas evalúan el mismo constructo y tienen igual peso psicológico en la mente del participante, lo que permite hallar la valoración global del cuestionario simplemente sumando las puntuaciones de cada ítem individual. El número de respuestas a cada pregunta debe ser impar para que se pueda recoger la opción de indiferencia. Para que la confianza en el resultado del cuestionario sea alta, es necesario que éste sea válido y fiable. La validez es una medida psicométrica del cuestionario que garantiza que "mide lo que realmente se quiere medir" (Paz 1996). La fiabilidad es otra medida que garantiza que las mediciones de un sujeto a otro y de un momento a otro y bajo premisas similares son consistentes (Muñiz 1996). Los cuestionarios de actitud aplicados como método de



evaluación basado en investigación permiten obtener medidas relativas a la percepción que los usuarios tienen de su propia eficiencia utilizando la interfaz, el grado en el cual la interfaz les gusta o disgusta, cuán útil perciben los usuarios que es el sistema, qué seguridad tienen los usuarios en la interacción con la interfaz, y el grado en el que los usuarios creen que aprenderían más acerca del sistema gracias a su interfaz.

En cuanto al coste de la técnica, varía dependiendo del uso que se haga de ella. En (Mayhew & Mantei 1994) se realiza una estimación del presupuesto que hay que destinar a usabilidad en un desarrollo realizado siguiendo las fases de ciclo de vida propuesto en (Mayhew 1999). Teniendo en cuenta que en dicha estimación se presuponen métodos de test de usabilidad para la evaluación en cada uno de los estados del sistema, y que habitualmente los cuestionarios utilizados en este tipo de pruebas no responden a las medidas de validez y fiabilidad más deseables, aspecto que abarata los costes (Brooke 1996), el presupuesto destinado a la elaboración, ejecución y análisis de datos sería el 4,6 % del coste total, según la estimación que hace María Elena García Barriocanal en su tesis, de la cual se está extrayendo toda esta información. En caso de realizar evaluación heurística, disminuiría dicho coste debido a que la fase de preparación del cuestionario es más corta: los ítems del cuestionario son los errores encontrados y, al no ser un cuestionario de actitud, no hay obtener medidas en cuanto a su validez y fiabilidad.

2.2.1. Cuestionarios de usabilidad existentes

Una de las principales ventajas de los cuestionarios es que una vez que se han diseñado pueden ser reutilizados para evaluar la mayoría de las interfaces de usuario. Esta característica hace que estén disponibles cuestionarios comerciales de probada fiabilidad y eficiencia. La adquisición de estos cuestionarios permite ahorrar en algunos casos el coste de diseñar otro validado y fiable. Incluso, es habitual que en la mayoría de ellos se indique el número de participantes que deben responder al cuestionario para proporcionar resultados significativos.

Los cuestionarios suelen estar orientados a un tipo de interfaz de usuario concreta: interfaces Web, interfaces de aplicaciones de consola o interfaces de sistemas multimedia. Algunos de los cuestionarios preparados para realizar evaluación basada únicamente en cuestionarios más habituales son: *QUIS*, *PUEU*, *CSUQ*, *PUTQ*, *SUMI* e *IsoMetrics*, que están orientados a evaluar interfaces de usuario de software de consola, *MUMMS*, orientado a evaluar interfaces de usuario en aplicaciones multimedia y *WAMMI*, especialmente diseñado para evaluar interfaces Web.



Cuestionario QUIS:

QUIS v5. 5 (*Questionnaire for User Interface Satisfaction*) (Chin et al. 1988) Se trata de un cuestionario Likert de 27 cuestiones con una escala para cada respuesta de uno a nueve y con las puntuaciones de los todas ellas convenientemente adjetivadas. Las cuestiones que lo componen están divididas en cinco secciones, cada una de las cuales destinada a obtener la evaluación de un aspecto concreto:

1. Reacción general al software, compuesta por seis preguntas.
2. Pantallas, compuesta por cuatro preguntas.
3. Terminología e información sobre el sistema, compuesta por 6 preguntas.
4. Aprendizaje, compuesta por seis preguntas.
5. Capacidades del sistema, formada por 5 preguntas.

Cuestionario PUEU:

El cuestionario PUEU (*Perceived Usefulness and Ease of Use*) (Davis 1989) está formado por 12 cuestiones Likert, originariamente escaladas de 1 a 7. Las cuestiones se dividen en dos secciones: Percepción de la utilidad, que abarca los seis primeros ítems, y percepción de facilidad de uso, que se compone de las seis restantes.

Cuestionario CSUQ:

CSUQ (*Computer System Usability Questionnaire*) (Lewis 1995) es también un cuestionario Likert de 19 cuestiones valoradas de uno a siete. Las cuestiones están divididas en tres secciones (no explícitamente mostradas, como ocurría en los casos anteriores), la primera de ellas, compuesta de ocho preguntas, está orientada a medir la utilidad del sistema, recoge además datos acerca de la satisfacción general del usuario, la eficiencia y la facilidad de aprendizaje. La segunda sección se orienta a la calidad de la información (7 cuestiones) y la tercera, a la calidad de la interfaz (4 preguntas). El mismo autor ha desarrollado un cuestionario de satisfacción post-tarea, denominado ASQ (*After-Scenario Questionnaire*) compuesto de 3 ítems, que constituye uno de los pocos cuestionarios de este tipo con fiabilidad probada.

Cuestionario PUTQ:

PUTQ (*Purdue Usability Testing Questionnaire*) (Lin et al. 1997) está compuesto de cien preguntas bipolares (ausencia - presencia) con una escala de uno a siete, divididas en ocho secciones:

1. Compatibilidad, bien sea entre la captura de datos y la información necesaria para introducirla o respecto a la familiaridad de la



información textual con convenciones habituales. Esta sección está compuesta de cinco preguntas.

2. Consistencia, compuesta por 17 cuestiones.
3. Flexibilidad, compuesta por 14 cuestiones.
4. Capacidad para ser aprendida fácilmente, formada por 7 preguntas.
5. Eficiencia, compuesta por 13 cuestiones.
6. Facilidad para ser recordada, formada por 15 preguntas.
7. Limitación perceptual, compuesta por 17 preguntas.
8. Control de manejo por parte del usuario, formada por las 12 cuestiones restantes.

Este cuestionario no está orientado a medir la satisfacción general del usuario ante un determinado sistema, sino que propone implícitamente una revisión de las principales directrices que hacen que una aplicación sea usable.

Cuestionario SUMI:

El cuestionario SUMI (*Software Usability Measurement Inventory*) (Kirakowski 1996) es uno de los cuestionarios Likert más utilizados para medir la calidad del software desde el punto de vista del usuario. Está compuesto por 50 preguntas con tres posibles valores de respuesta que se dividen en cinco secciones:

1. Eficiencia, compuesta de 10 cuestiones para medir la sensación que tienen los usuarios respecto a si el software les ayuda a realizar su trabajo.
2. Satisfacción, compuesta por 10 preguntas que miden la reacción emocional general que el software provoca en el usuario.
3. Control, se compone de 10 preguntas que miden si el usuario se siente seguro de sus acciones cuando utiliza la aplicación.
4. Facilidad de aprendizaje, que mide la percepción del usuario sobre si es fácil aprender manejar el sistema y aprender a utilizar nuevas funcionalidades del mismo cuando sea necesario, utilizando para ello 10 preguntas.
5. Utilidad, que se compone de 10 cuestiones para evaluar la calidad de la ayuda y documentación del sistema así como si la interfaz es autoexplicativa.

Cuestionario IsoMetrics:

El cuestionario IsoMetrics (Gediga et al. 1999) está disponible en dos versiones, una denominada larga (*IsoMetricL*) y otra denominada corta (*IsoMetricsS*). Ambas se componen de 75 ítems Likert que se pueden valorar de uno a cinco, y se diferencian en que las cuestiones de la versión IsoMetricsL van acompañadas tanto de otra pregunta que permite expresar al usuario su opinión (de uno a cinco también) acerca de si el ítem evaluado



es importante para su apreciación global del software, como de una pregunta abierta en la que el usuario tiene que ejemplificar qué le ha llevado a dar poca puntuación al ítem (si es que lo ha hecho así). Las preguntas que los componen se dividen en siete secciones:

1. Adecuación de la tarea (15 elementos).
2. Autodescripción (12 elementos).
3. Capacidad de control por parte del usuario de la aplicación (11 elementos).
4. Conformidad con las expectativas del usuario (8 elementos).
5. Tolerancia de error (15 elementos).
6. Adecuación para la individualización (6 elementos).
7. Adecuación para el aprendizaje (8 elementos).

Cuestionario MUMMS:

En cuanto a la evaluación de interfaces multimedia, el cuestionario más conocido es MUMMS2.0 (*Measuring the Usability of Multimedia Software*), creado por el mismo grupo de investigación que SUMI (*HFRG, Human Factors Research Group*). Este cuestionario Likert con tres posibles valores para cada pregunta está compuesto de 60 cuestiones, divididas de diez en diez en seis grupos, cinco de ellos iguales que en SUMI (eficiencia, satisfacción, control, facilidad de aprendizaje y utilidad), y una sexta, entusiasmo, que mide el grado en el que los usuarios se sienten inmersos en la aplicación multimedia.

Cuestionario WAMMI:

El cuestionario WAMMI2.0 (*Website Analysis and Measurement Inventory*) (Kirakowski & Cierlik 1998), creado también por el grupo *HFRG*, permite llevar a cabo evaluaciones de usabilidad de sitios Web mediante 20 cuestiones con escala Likert, divididas en cinco secciones:

1. Atractivo del sitio.
2. Control percibido por usuario durante la navegación.
3. Eficiencia para realizar la tarea deseada.
4. Utilidad de la página o sitio Web.
5. Facilidad con la que se puede aprender el manejo y navegación del sitio.

Cabe destacar que cada ítem actúa de manera ponderada en la puntuación obtenida en cada una de las secciones del cuestionario, y un mismo ítem se puede tener en cuenta para obtener la puntuación en más de una sección.

A pesar de ser muchos los cuestionarios disponibles para realizar evaluación de usabilidad, no existe ninguna especificación estandarizada a



la cual acogerse en el diseño de un nuevo cuestionario, siendo la única conocida la creada por el consorcio IMS, denominada *QTI Question and Test Interoperability* (Almond 2002). El ámbito de esta especificación es exclusivamente la definición de cuestionarios de evaluación de conocimientos, con objeto de que puedan ser intercambiados entre distintas plataformas de sistemas de gestión de aprendizaje (LMS); si bien, a pesar de tener ciertas carencias (García & Sicilia 2002), su ámbito de aplicación puede ser ampliado para definir otro tipo de cuestionarios aplicables en dichos entornos (García, Sicilia, Hilera & Gutiérrez 2002).

2.2.2. Herramientas para la creación de cuestionarios

Además de los cuestionarios de usabilidad mencionados en la sección anterior, hay actualmente diferentes sistemas basados en tecnología Web que facilitan el diseño de un nuevo cuestionario o la utilización de alguno de los ya existentes.

La herramienta propuesta en (Perlman 2002) permite la edición de cuestionarios existentes. Es una herramienta gratuita, basada en el lenguaje de programación Web Perl, con la que el evaluador puede seleccionar un cuestionario o heurística que estará disponible a través de Web para que los participantes en la evaluación lo respondan y envíen automáticamente por correo electrónico a la dirección del encargado de la evaluación. Esta herramienta permite personalizar la prueba indicando unos datos específicos.

Con el fin de personalizar los ítems que forman el cuestionario, en alguno de ellos se indica el nombre del objeto a evaluar (dependiendo del cuestionario seleccionado) y además:

- ❖ Dirección de correo electrónico a la que se deben enviar las respuestas.
- ❖ Cuestionario que se quiere utilizar, estando disponibles los cuestionarios QUIS, PUEU, CSUQ, ASQ y PUTQ y las heurísticas NHE y PHUE.
- ❖ Valores máximo y mínimo que se deben mostrar en la escala Likert para cada cuestión del test.
- ❖ Etiquetas que se quieren mostrar junto con los valores máximo y mínimo de la escala Likert.



- ❖ Etiqueta que se desea mostrar para el valor que indica que la pregunta no es aplicable al sistema evaluado, por ejemplo, "No Aplicable", y el valor que se enviará por correo en caso de seleccionar esta opción, por ejemplo "na".
- ❖ Se debe indicar el número de líneas permitidas para los comentarios que pueda realizar el usuario.
- ❖ También se permite incluir al final del cuestionario algunas líneas para que el participante indique mediante respuestas abiertas los aspectos positivos y negativos que tiene el sistema. Es necesario personalizar en cada cuestionario el número de opiniones que se pueden realizar a este respecto.

El usuario que responda el cuestionario lo hará a través de un navegador, y tendrá que indicar su nombre o el identificador que se le haya asignado. Sus respuestas serán enviadas al evaluador encargado de la prueba junto con el nombre del participante y la fecha y hora en la que se llevo a cabo cuando el participante pulse el botón *Mail Data* en la propia interfaz del cuestionario. El evaluador recibirá en el correo electrónico las respuestas al cuestionario como pares (*número de pregunta = valor seleccionado*).

Otra herramienta Web de generación de cuestionarios es WhizQuest v1.0, desarrollada por el grupo ISIS (*Information Systems and Insect Studies*) del Instituto Internacional de Tecnologías de la Información de la Universidad Politécnica de Virginia (ISIS Group 2002). Dado que la herramienta, a diferencia de la anterior, no está desarrollada para diseñar exclusivamente cuestionarios de satisfacción, permite que en los nuevos cuestionarios se puedan establecer ítems con diferentes tipos de respuestas (numéricas, de respuesta abierta, con escala y de múltiple opción). Además de la generación de la prueba, la herramienta genera informes con los diferentes resultados de los cuestionarios. Dependiendo del tipo de pregunta respondida se pueden elaborar informes a partir de los siguientes datos:

- ❖ Cuando los ítems del cuestionario se responden con los valores de una escala, se obtiene para cada pregunta el número de veces que se ha respondido cada valor posible junto con la media de todos los cuestionarios para la pregunta y la desviación típica. También se indica el porcentaje de participantes que respondieron a la pregunta y si el dato ha sido incluido ya en un informe.
- ❖ Cuando las preguntas son de respuesta abierta, se muestra el texto escrito por cada participante, indican también su nombre y hora en la que introdujo el comentario. Además, también se incluye un indicador



que informa sobre si la respuesta ya ha sido incluida en un informe de resultados.

- ❖ Las preguntas dicotómicas y de múltiple opción se utilizan para recabar información y su resultado sirve para configurar un fichero de historia, añadiendo además de manera automática en el fichero la fecha y hora en la que el participante introdujo los datos y el nombre del ordenador desde el que realizó el cuestionario.

Para cualquier pregunta, independientemente de su tipo, el administrador del cuestionario tiene acceso a los datos sin procesar.

Además de estas herramientas para la creación de cuestionarios, desarrolladas por grupos de investigación, existen otras muchas en el mercado comercializadas por empresas del sector de Tecnologías Educativas o de Interacción Persona-Ordenador, como *Experience Evaluator*, de *NetRacker*.

3. SOLUCIÓN DEL PROBLEMA

3.1. Arquitectura de la solución

La arquitectura utilizada para el diseño de la aplicación es una arquitectura **Web**. En este modo se dispone de tres capas. La capa de los **clientes**, la capa del **servidor** de aplicaciones y la capa de la **base de datos**.

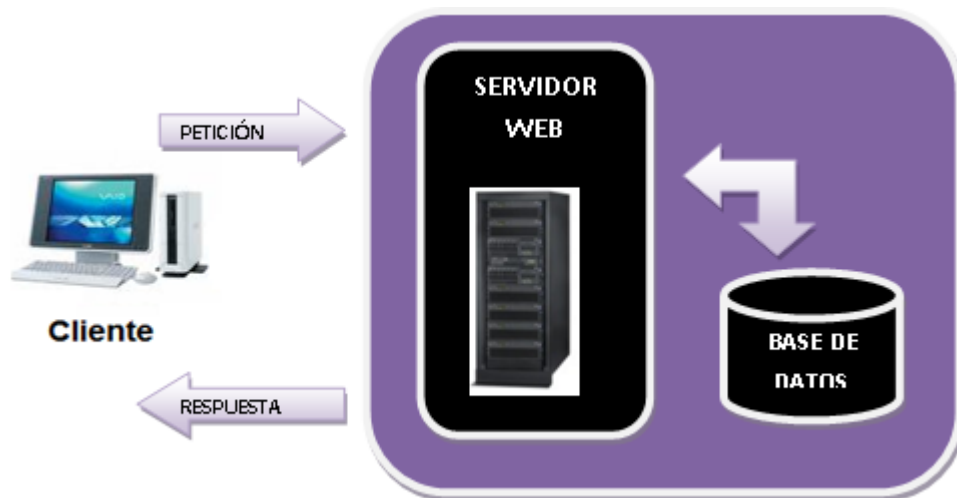


Figura 1. Arquitectura de la solución

En la capa de cliente se sitúan como su nombre indican todos las máquinas cliente de la aplicación. Los clientes no contienen nada de la lógica de negocio de la aplicación. Toda la lógica de la aplicación está situada en la segunda capa o nivel, es decir, en el servidor de aplicación y en la base de datos se almacena toda la información que maneja la aplicación de manera permanente.

Cabe destacar, que los puestos de los clientes utilizan un modelo de cliente ligero. Debido a que toda la lógica de la aplicación se encuentra situada en el servidor de aplicaciones, los puestos de los clientes solo deben tener el software necesario para que el usuario pueda invocar la funcionalidad del servidor de aplicaciones y pueda visualizar el resultado de la ejecución. Debido a que se ha utilizado el **protocolo HTTP**, este software será un navegador.

Como es lógico, las comunicaciones entre las tres capas variarán dependiendo de las capas que se comunican. La comunicación entre los puestos de clientes y el servidor de aplicaciones se realizará a través de

Internet. En cuanto a la comunicación del servidor de aplicaciones con la base de datos se realizará mediante una red interna de la entidad.

En cuanto al patrón arquitectónico, la aplicación sigue fielmente el patrón MVC (Modelo-Vista-Controlador).

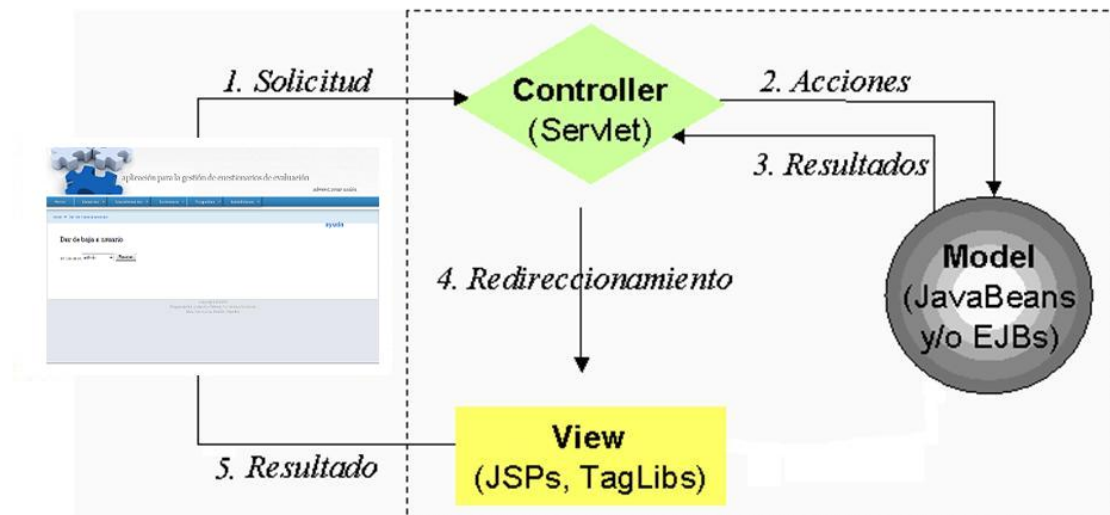


Figura 2. Patrón MVC

El patrón de arquitectura MVC (Model-View-Controller) más que un patrón es una filosofía de diseño de aplicaciones en la que se define la organización independiente del Model (modelo, formado por los Objetos de Negocio), la View (vista, interfaz con el usuario u otro sistema) y el Controller (controlador del flujo de trabajo de la aplicación).

La sección 3.2.2. *Struts* [9] añade algunos aspectos acerca del patrón arquitectónico MVC.

3.2. Tecnologías utilizadas

Esta sección describe las tecnologías empleadas para el desarrollo del proyecto. La plataforma sobre la que se ha trabajado para el desarrollo de la aplicación y sobre la que están basadas el resto de tecnologías empleadas es Java [5].

3.2.1. Java

Los lenguajes estructurados se basan en estructuras de control de bloques de código y subrutinas independientes que soportan recursividad y variables locales. La programación orientada a objetos coge las mejores



ideas de la programación estructurada y las combina con nuevos conceptos de organización, permitiendo descomponer un programa en grupos relacionados. Cada subgrupo pasa a ser un objeto autocontenido con sus propias instrucciones y datos.

El lenguaje de programación empleado para el modelo y controlador de la aplicación ha sido Java.

Sun describe el lenguaje Java como “simple, orientado a objetos, distribuido, interpretado, robusto, seguro, de arquitectura neutra, portable, de altas prestaciones, multitarea y dinámico”.

Nótese “de arquitectura neutra”, con Java se desarrolló un código “neutro” que permitía ser ejecutado sobre una “máquina hipotética o virtual”, denominada Java Virtual Machine (JVM). La JVM es la que interpreta este código neutro convirtiéndolo a código particular de la plataforma sobre la que se trabaja.

La JVM es el intérprete de Java. Ejecuta los *bytecode* (archivos con extensión *.class) creados por el compilador de Java (javac.exe).

En la Figura 3 se muestra un esquema de tres ediciones de plataformas diferentes desarrolladas por Java:

- 🚦 **J2SE** → *Java2 Standard Edition*. Es la más utilizada en Java, contiene un conjunto de ejecución y de APIs para crear aplicaciones no destinadas a ser centralizadas y de accesos concurrentes.
- 🚦 **J2EE** → *Java2 Enterprise Edition*. Es una plataforma para crear aplicaciones de servidor. Reúne un conjunto de especificaciones Java orientadas al desarrollo de aplicaciones empresariales.
- 🚦 **J2ME** → *Java2 Micro Edition*. Permite la creación de aplicaciones Java para “micro-dispositivos” con un apoyo de pantalla y memoria limitada, como pueden ser teléfonos, PDAs,... etc. Algunas de las tecnologías que soporta son: *Bluetooth*, *J2ME Web services*, *JavaTV*, *JavaPhone*, *J2EE client* o *Java Card*.



Figura 3. Clasificación de la Plataforma Java

J2EE se materializa a través de un conjunto de especificaciones, cada una de las cuales cumple un papel concreto en el puzzle global de las aplicaciones empresariales: *JSP*, *Servlet*, *EJB*, *JDBC*, *JavaMail*, *JMS* o *JAXP* entre otras.

Para el desarrollo de la aplicación se ha hecho uso de las especificaciones *JSP*, *Servlet*, *EJB* y *JDBC*. A continuación se explica brevemente cada una de ellas:

✚ **JSP** es una interfaz de programación de aplicaciones de servidores Web. En una página jsp se entremezclan bloques de HTML estáticos, y HTML dinámico generados con Java que se ejecutan en el servidor. Una página jsp puede procesar formularios Web, acceder a bases de datos y redireccionar a otras páginas. Las páginas jsp son transformadas a un Servlet y después compiladas. El contenedor JSP proporciona un motor que interpreta y procesa una página JSP como un Servlet (en *Tomcat*, dicho motor se llama *Jasper*). Al estar basadas en los Servlets, las distintas peticiones a una misma página jsp son atendidas por una única instancia del Servlet.

Una página JSP es básicamente una página Web con HTML tradicional y código Java incrustado. La extensión de fichero de una página JSP es ".jsp" en vez de ".html" o ".htm", y eso le indica al servidor que esta página requiere un tratamiento especial que se conseguirá con una extensión del servidor o un *plug-in*.

El servidor web comprueba si la página ha sido solicitada con anterioridad. En ese caso el Servlet correspondiente ya estará cargado en memoria. Si no es así, se notifica al motor de jsp y se generará un Servlet para la página.

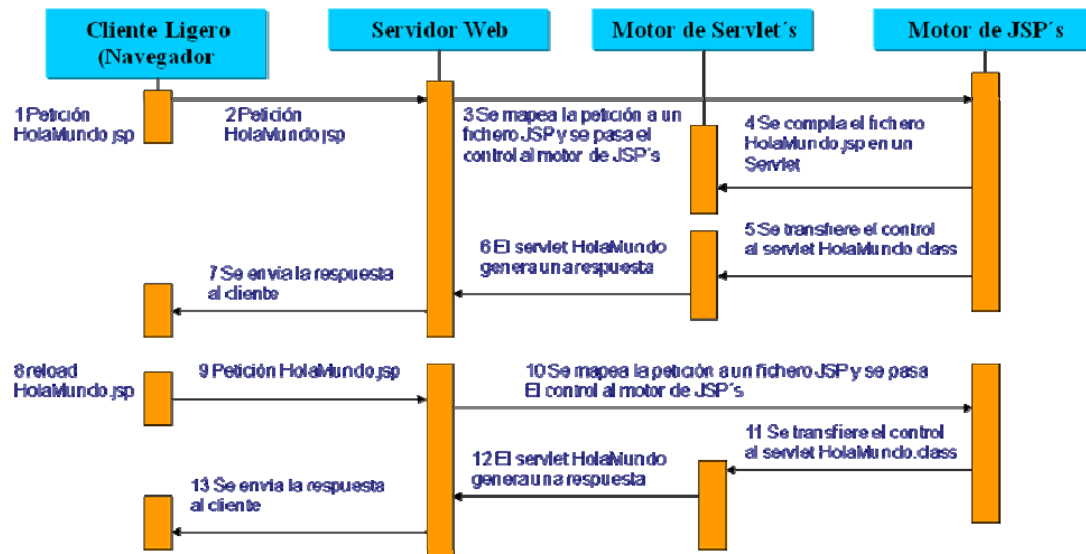


Figura 4. Funcionamiento de la especificación JSP 1.2

Cuando un cliente solicita una página jsp, se ejecuta en el servidor el código JSP de la página, dando como resultado una página HTML que se fusiona con el HTML original, generando una página HTML de respuesta que será enviada al cliente.

- Los **Servlets** son, en una aplicación J2EE, objetos destinados a gestionar el flujo de peticiones, delegando la responsabilidad de lógica de negocio en *Beans* o EJB (*Enterprise Java Beans*), y posteriormente enviar la respuesta a las vistas (jsp) para que generen la respuesta al usuario. Dicho de otra forma, están diseñados para ofrecer contenido dinámico desde un servidor web, y controlar el flujo de control de las aplicaciones web.

Entre el servidor de aplicaciones (o *web container*) y el Servlet existe un contrato que determina cómo han de interactuar. La especificación de éste se encuentra en los JSR (*Java Specification Requests*) del JCP (*Java Community Process*).

Los Servlet han sido especialmente diseñados para trabajar dentro de un modelo de proceso petición/respuesta (el cliente manda una petición [http, https, ..] al servidor y éste procesa y devuelve una respuesta). Por consiguiente, los Servlet son programas que se ejecutan en un contenedor web y actúan como intermediarios entre la petición del cliente y el procesamiento de la información y posterior respuesta.

Los contenedores de Servlet son los responsables de manejar las peticiones de los clientes, proporcionárselas al Servlet invocado y devolver la respuesta al cliente. La interfaz entre los contenedores y los Servlet viene especificada por la API de Servlet, que especifica los métodos que debe tener un Servlet y como estos serán invocados por el contenedor de Servlet, así como las clases de objetos destinados a dicha comunicación.

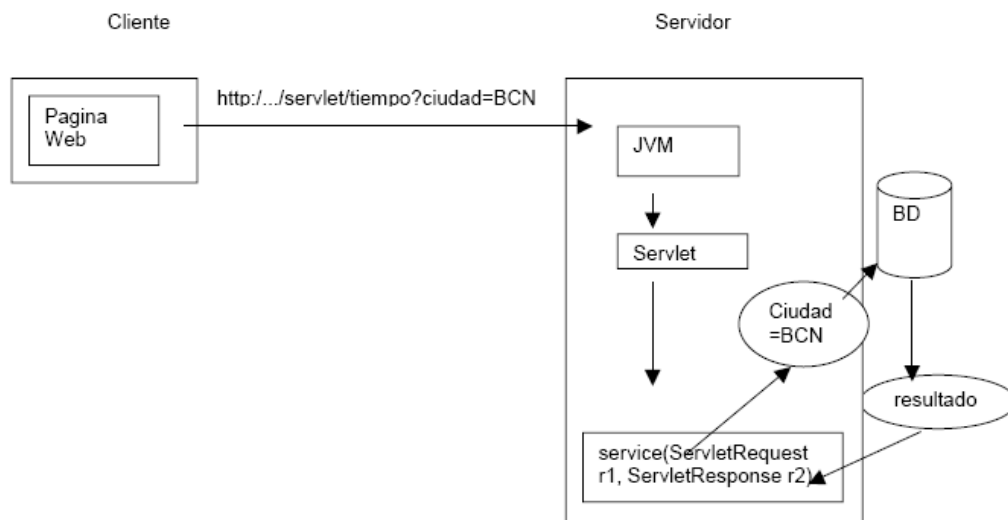


Figura 5. Funcionamiento de un Servlet genérico

- Los **EJBs** son una colección de clases Java y un archivo *XML*, integrados en una única unidad. Estas clases Java siguen algunas reglas y proporcionan métodos específicos de retrollamada.

Un EJB se ejecuta en un contenedor (véase Figura 6), que a su vez se ejecuta en un servidor de aplicaciones y se responsabiliza de los temas a nivel de sistema. Esto permite al desarrollador centrarse en la lógica de empresa, este es el pensamiento subyacente de la tecnología EJB.

Los EJB son tecnología Java, por lo que no están ligados a ningún Sistema Operativo.

Un EJB pretende ser un lote reutilizable de lógica de empresa. Pueden funcionar con cualquier tipo de cliente, tanto Web (combinados con JSPs y Servlets) como Java (que utilicen RMI/IIOP) entre otros.

La especificación EJB tiene como intención dar una especificación de componentes del lado del servidor (dentro del entorno J2EE).

La arquitectura JavaBeans está diseñada para proporcionar un formato para componentes de fines generales, mientras que la arquitectura EJB proporciona un formato para componentes de lógica de empresa altamente especializados desplegados en un servidor J2EE.

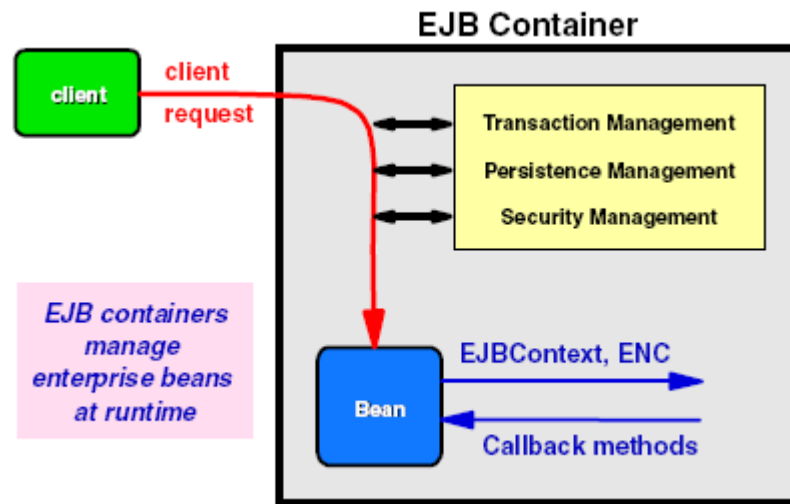


Figura 6. Contenedor de EJB

Actualmente el estándar EJB define tres tipos de EJB:

- **Message-driven Beans:** El MDB es esencialmente un oyente de mensajes, que puede consumir mensajes de una cola o de una suscripción duradera a través del contenedor J2EE: *"El controlador invoca al bean como resultado de la llegada de un mensaje"*.
- **Session Beans:** Los EJB de sesión se encargan de resolver la lógica de negocio de la aplicación. Dentro de la especificación se pueden encontrar dos tipos de Session Beans: *State Full (SBFS)* que son bean de sesión con estado y pueden mantener datos entre diferentes accesos de un cliente y *State Less (SBSL)* que son bean de sesión sin estado y no mantienen dichos datos.
- **Entity Beans:** Los EJBs de entidad representan el modelo de datos de la aplicación. Están directamente relacionados con los datos de la aplicación, son objetos que mantienen en memoria los datos que maneja la aplicación.



Los bean de sesión pueden acceder también a los datos, pero a diferencia de los bean de entidad, no pueden proporcionar una representación orientada a objetos de los mismos.

Los bean de entidad son diferentes porque su estado es almacenado en la base de datos, varios clientes pueden por tanto, acceder a su estado simultáneamente.

Proporcionan además una serie de servicios de retrollamada a nivel de sistema como son: información del progreso transaccional actual, compatibilidad con accesos concurrentes, mantenimiento de caché entre peticiones y proporcionar todo el código de gestión de persistencia.

✚ **JDBC** (*Java DataBase Connectivity*) es un API de Java que permite al programador ejecutar instrucciones en lenguaje estándar de acceso a Bases de Datos, SQL (*Structured Query Language*, lenguaje estructurado de consultas), que es un lenguaje de muy alto nivel que permite crear, examinar, manipular y gestionar bases de datos relacionales. Para que una aplicación pueda hacer operaciones en una base de datos, ha de tener una conexión con ella, que se establece a través de un driver, que convierte el lenguaje de alto nivel a sentencias de base de datos. Es decir, las tres acciones principales que realizará JDBC son las de establecer la conexión a una base de datos, ya sea remota o no; enviar sentencias SQL a esa base de datos y, en tercer lugar, procesar los resultados obtenidos de la base de datos. (Para más información ver sección 3.2.4. *MySQL y JDBC* del documento).

3.2.2. Struts

Struts [9], nacido en mayo del año 2000 en el seno de la *Apache Software Foundation*, consiste en un entorno de desarrollo o *framework* para desarrollar aplicaciones web que implementan el patrón Modelo Vista Controlador en el lenguaje Java.

Un framework o marco de desarrollo es un término que se utiliza para definir a una serie de clases, interfaces, módulos, y librerías que cooperan conjuntamente para solucionar un determinado problema.

Para la implementación del patrón MVC se ha utilizado Struts.

Una aplicación basada en Struts, tiene un componente básico llamado *ActionServlet*. Este es un *Servlet* que tramita las peticiones de los clientes delegando a un componente definido por el usuario por cada petición.

Este Servlet es el punto central del framework, aunque no es necesario que toda la actividad fluya a través de él. En una aplicación basada en Struts se pueden hacer peticiones a una JSP que contengan o no *tag libraries* de Struts, sin pasar por el Servlet ActionServlet.

El ActionServlet (controlador) de Struts captura y encamina las peticiones HTTP que llegan a la aplicación (toma la decisión de a dónde enviar la petición HTTP), a otros componentes de aplicación. Estos componentes pueden ser páginas JSP o instancias de una subclase de la clase *org.apache.struts.action.Action* que el propio framework suministra.

Cuando se inicia el Servlet ActionServlet, carga y analiza la información de un fichero que contiene la configuración de la aplicación para aplicar las características de Struts. Entre otras cosas, el fichero de configuración define las correspondencias que existen entre las peticiones HTTP que captura el Servlet controlador y las acciones que van a tratar esa petición.

Estas correspondencias son manipuladas como instancias de la clase *org.apache.struts.action.ActionMapping*

El funcionamiento de Struts es sencillo, el navegador lanza una petición HTTP a la aplicación, éste evento es capturado por el servidor de aplicaciones y encaminado al componente correspondiente para su tratamiento. Véase Figura 7:

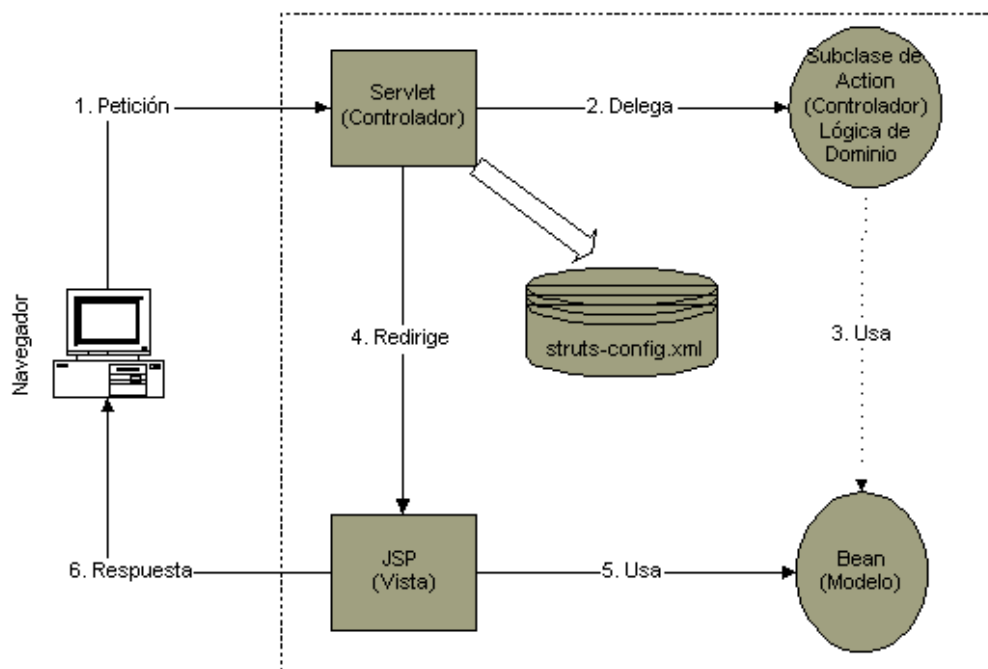


Figura 7. Servidor de aplicaciones



Controlador: el Servlet `ActionServlet` actúa de controlador. Para ello recibe la petición del navegador y decide qué subclase de `Action` va a tratar la petición en función de lo que se ha declarado en el fichero de configuración *struts-config.xml*.

Por otro lado la subclase de `Action`, actualiza el estado del modelo, y, controla el flujo de la aplicación y tratamiento de errores. Una instancia de una subclase de `Action` puede tratar la petición y responder al cliente o indicar al Servlet controlador a qué componente del sistema debe delegar el control (esta es la opción la que se lleve a cabo). Las instancias de las subclases de `Action` tienen acceso al contexto del Servlet controlador y demás objetos que actúan con el contenedor Web.

Modelo. Representa al estado de la aplicación. Estará formado por dos tipos de clases:

- ✚ Subclases de la clase base *org.apache.struts.action.ActionForm* que proporciona Struts. Se utiliza para obtener la entrada de datos proporcionada por el usuario en la petición HTTP.
- ✚ Clases ordinarias sin necesidad de extender `ActionForm`.

Vista. La vista estará formada por páginas JSP que no contendrán lógica de negocio, ni flujo de la aplicación e información del modelo, sólo *tags*. Utiliza el modelo generado para obtener la información y presentarla.

3.2.3. Hibernate

Con el fin de mantener la persistencia de la aplicación se utiliza el motor de persistencia **Hibernate [3]**.

Hibernate [3], al ser un motor de persistencia, nos facilitará la traducción entre dos formatos de datos, de registros a objetos y de objetos a registros. Para ello crearemos dos tipos de archivos:

- ✚ **El archivo de propiedades de Hibernate** (*Hibernate.properties* / *hibernate.cfg.xml*) que contiene el gestor de base de datos a usar (MySQL), y a qué base de datos debe conectar (en este caso **bd_gce**) y cómo.
- ✚ **Los archivos que definen el emparejamiento** (*mapping*) de propiedades con tablas y columnas (**.hbm.xml*). Se tiene uno por cada clase, con el nombre *NombreClase.hbm.xml*, que describe cómo se relacionan clases y tablas y propiedades y columnas.

La arquitectura de Hibernate [3] queda descrita en la siguiente imagen:

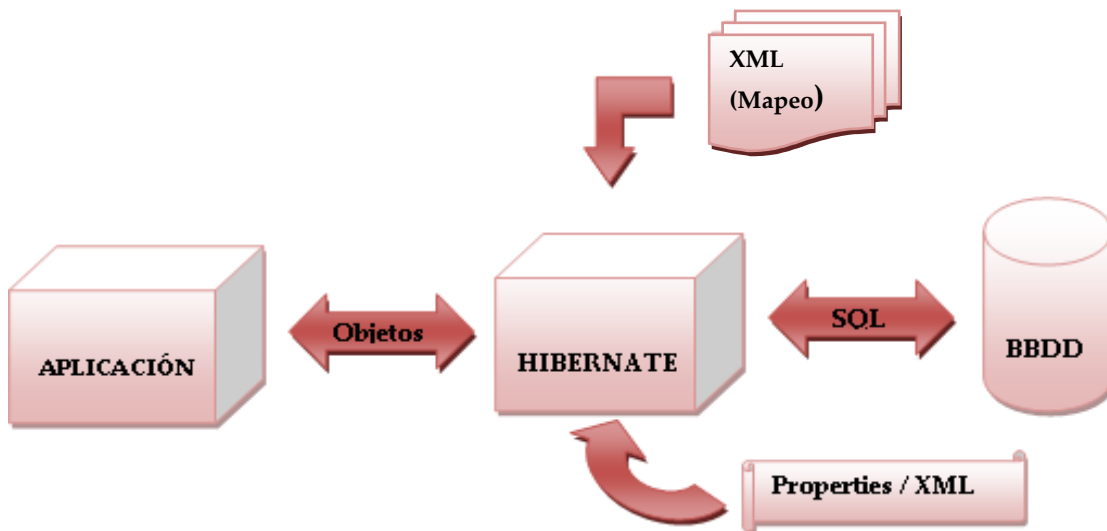


Figura 8. Arquitectura Hibernate

3.2.4. MySQL y JDBC

El sistema gestor de bases de datos utilizado es **MySQL** [8].

Se tiene por tanto, una base de datos relacional. Una base de datos relacional representa un conjunto de datos que están almacenados en tablas entre las cuales se establecen unas relaciones para manejar los datos de una forma eficiente y segura. Para usar y gestionar una base de datos relacional se usa el lenguaje estándar de programación SQL.

MySQL es *Open Source*, lo que significa que el código fuente se puede descargar y está accesible a cualquiera, por otra parte, usa la licencia GPL para aplicaciones no comerciales.

Además MySQL proporciona rapidez, seguridad y facilidad de manejo. Gracias a la colaboración de muchos usuarios, la base de datos se ha ido mejorando optimizándose en velocidad. Por eso es una de las bases de datos más usadas en aplicaciones web.

Para las operaciones sobre la base de datos MySQL desde Java, se ha utilizado el API de Java **JDBC**.

JDBC es un API de Java para ejecutar sentencias SQL. Está formado por un conjunto de clases e interfaces programadas con el propio Java. Permite interactuar con bases de datos, de forma transparente al tipo de la misma. Es decir, es una forma única de programar el acceso a bases de datos desde Java, independiente del tipo de la base de datos. JDBC realiza llamadas directas a SQL. Para más información sobre JDBC se puede consultar la dirección <http://www.java.sun.com/products/jdbc/>.

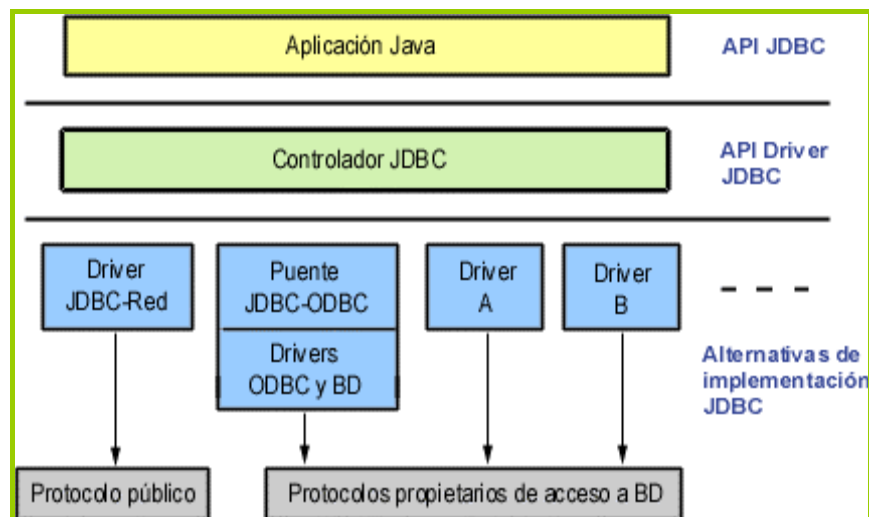


Figura 9. Drivers de JDBC

En la Figura 9 se esquematizan las cuatro categorías de drivers que soportan la conectividad JDBC: puente JDBC-ODBC, drivers de red y drivers nativos.

Para el presente proyecto se ha utilizado un driver específico que permite la comunicación directa de JDBC con MySQL. Su nombre es **MySQLConnector/J** y es un driver nativo que convierte llamadas JDBC al protocolo de red utilizado por la base de datos MySQL.

3.2.5. JUnit

Para la realización de las pruebas unitarias se ha utilizado la herramienta *JUnit* [7] en su versión 4.5.

Las pruebas unitarias ofrecen un camino para mantener en lista los posibles errores (*bugs*). Primero, esto aplica a piezas individuales de código -típicamente funciones o métodos- y a menos que estas pequeñas piezas sean correctas, es seguro que la edificación completa de software sea propensa a desmoronarse. Segundo, las pruebas unitarias permiten a las personas que escriben unidades de código probarlas también. Las pruebas



arrojan los errores, pero los programadores son generalmente resistentes a invertir tiempo a cualquier cosa excepto codificarlos.

Kent Beck y Erich Gamma son los creadores de JUnit, derivado del *framework* de pruebas para *Smalltalk*. La idea es probar y codificar simultáneamente. Los programadores escriben algunas líneas de código, entonces escriben la prueba para ello; si la prueba falla, ellos corrigen el error. Si todo está bien, ellos escriben un poco más de código y hacen más pruebas y así sucesivamente. Todas las pruebas son mantenidas junto con el código, entonces si el código funcional cambia, es fácil realizar pruebas anteriores para asegurarse que los cambios no rompieron el software.

Una fácil integración de código y pruebas unitarias recaen en el corazón de la Programación Extrema (*Extreme Programming*) y el movimiento de desarrollo de software ágil. En la programación ágil el software es construido mediante la implementación de funcionalidad incremental, en pequeños esfuerzos de actividad. Para realizar esto se requiere garantizar que lo que estamos implementando es correcto entonces podemos entregarlo con confianza y continuar construyéndolo.

En las pruebas unitarias, se ejemplifica algo de código, proporcionando entrada de datos, si es necesario, y examinando el comportamiento del código, usualmente en la forma de su salida. Una forma simple de hacer esto, y una que los programadores han usado ampliamente, emplea una serie de sentencias *If* que compara algo con el valor esperado, En cambio JUnit, en lugar de tratar sentencias *If*, escribe afirmaciones. Una afirmación es una forma de especificar el resultado deseado y comprarlo con el resultado actual. Si los resultados deseado y actual son idénticos, la afirmación es exitosa, de lo contrario, significará que se ha producido un error durante la ejecución del método.

Una prueba usualmente necesita de algunos soportes para inicializar variables y crear objetos. En JUnit, esta parte es llamada configuración o *setup*. Ésta es independiente de las afirmaciones, entonces el mismo soporte puede ser aplicado a varias pruebas individuales. La configuración debe ser realizada antes de cada prueba.

Cuando una prueba termina, el soporte puede requerir algunas acciones de limpieza. A este proceso en JUnit se le conoce como desmontaje (*teardown*). Éste asegura que cada prueba termina con un borrón y cuenta nueva. Entonces si otra prueba sigue a continuación, la configuración se ejecutará correctamente. La configuración y desmontaje son llamados como aparato de prueba.



Las pruebas individuales son llamadas casos de pruebas (*test cases*). Los casos de prueba raramente existen en un vacío. Si un proyecto asume las pruebas unitarias seriamente, este acumulará una gran colección de casos de prueba. Algunas veces, los programadores lo verán como ejercer algunos de estos casos juntos. Cuando esto ocurre, ellos pueden combinar los casos dentro de espacios de prueba (*test suites*) que pueden ejecutarse como un solo grupo.

Esta aplicación está compuesta por diferentes funcionalidades que se han ido verificando mediante casos de prueba individuales o *test cases* para cada una de ellas. Esto se ha hecho a través de la herramienta JUnit. Con estas pruebas, se ha pretendido verificar el buen funcionamiento de toda la lógica de negocio de la que se compone la aplicación, para agilizar la integración de ésta con la vista y el controlador, y de esta forma hacer más sencillas las pruebas de aceptación de la aplicación.

3.3. Entorno de desarrollo

Existen distintos programas comerciales que permiten desarrollar código Java. Varias compañías Sun, IBM,...etc., distribuyen gratuitamente el *Java(tm) Development Kit (JDK)*. Se trata de un conjunto de programas y librerías que permiten desarrollar, compilar y ejecutar programas en Java.

Existe también una versión reducida del JDK, denominada *JRE (Java Runtime Environment)* destinada únicamente a ejecutar código Java (no permite compilar).

Los *IDEs (Integrated Development Environment)*, son entornos de desarrollo integrados. En un mismo programa es posible escribir código y ejecutarlo sin tener que cambiar de aplicación. Algunos incluyen una herramienta para realizar *Debug* gráficamente, frente a la versión que incorpora el *JDK* basada en la utilización de una consola bastante difícil y pesada de utilizar. Estos entornos integrados permiten desarrollar las aplicaciones de forma mucho más rápida, incorporando en muchos casos librerías con componentes ya desarrollados, los cuales se incorporan al proyecto o programa. Podemos señalar como ejemplo las plataformas *WepSphere Studio* de IBM, *Forte* de Sun o *Eclipse* [2].

Para el desarrollo de la aplicación se ha utilizado como entorno Eclipse, más concretamente *MyEclipse 6.0.1*. *MyEclipse* es un *plug-in* de Eclipse.

IBM desarrolla inicialmente Eclipse [2] en el año 1999, pero hasta 2001 no nace la primera versión completa. Eclipse es una plataforma



implementada en Java cuya utilidad radica en integrar herramientas de desarrollo. Está basada en plug-ins.

Eclipse es mucho más que un editor, se trata de un IDE completo, con la posibilidad de añadirle plug-ins para casi todo lo que se necesite y poder escribir en más lenguajes que no sean Java como *PHP*, *Ruby*, *C++*,... hay tanto versiones basadas en el IDE de Eclipse original como plug-ins para añadirle a un mismo Eclipse todas las funcionalidades necesarias.

Una de las grandes ventajas de Eclipse es que al estar desarrollado en Java es multiplataforma.

Actualmente Eclipse es uno de los entornos más utilizados, en Internet hay gran cantidad de tutoriales que lo tratan y además es *open source*.



4. ANÁLISIS DE LA APLICACIÓN

4.1. Especificación de los requisitos de usuario

Tras varias reuniones con el cliente, se han extraído los siguientes requisitos de usuario:

4.1.1. Requisitos de capacidad

ID	RC1
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Gestión de usuarios</i>	
<p>La aplicación permitirá una gestión de usuarios que abarcará las funcionalidades de dar de alta a un usuario en la base de datos, darle de baja y consultar o modificar sus datos personales.</p> <p>Asimismo se tendrán dos tipos de usuarios:</p> <ul style="list-style-type: none"> • Usuario “básico” → podrá completar un cuestionario y modificar su contraseña de acceso a la aplicación. • Usuario administrador → podrá acceder a las funcionalidades del usuario básico y además podrá llevar a cabo una gestión de usuarios, gestión de cuestionarios, gestión de secciones, gestión de preguntas así como visualizar estadísticas basadas en los cuestionarios respondidos por los usuarios. 	

Tabla 1. RC1- Gestión de usuarios

ID	RC2
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Gestión de cuestionarios</i>	
<p>La aplicación permitirá una gestión de cuestionarios que abarcará las funcionalidades de insertar un cuestionario nuevo en la base de datos, eliminarlo y consultar o modificar sus datos.</p>	

Tabla 2. RC2- Gestión de cuestionarios



ID	RC3
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Gestión de secciones</i>	
La aplicación permitirá una gestión de secciones que abarcará las funcionalidades de insertar una nueva sección en la base de datos, eliminarla y consultar o modificar sus datos.	

Tabla 3. RC3- Gestión de secciones

ID	RC4
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Gestión de preguntas</i>	
La aplicación permitirá una gestión de preguntas que abarcará las funcionalidades de insertar una nueva pregunta en la base de datos, eliminarla y consultar o modificar sus datos.	

Tabla 4. RC4 Gestión de preguntas

ID	RC5
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Visualizar estadísticas</i>	
<p>La aplicación mostrará dos tipos de resúmenes o estadísticas basadas en las respuestas de los usuarios a la hora de completar los cuestionarios:</p> <ul style="list-style-type: none"> • Promedios de puntuaciones por pregunta y característica → A partir de un cuestionario, se mostrarán sus preguntas y la media aritmética de las puntuaciones dadas por pregunta y característica de la misma. • Porcentaje usuarios que han respondido cada opción por pregunta y característica → A partir de un cuestionario, se mostrarán sus preguntas y el porcentaje de usuarios que ha respondido cada opción por pregunta y característica. 	

Tabla 5. RC5- Visualizar estadísticas



ID	RC6
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Completar cuestionario</i>	
La aplicación permitirá a todos los usuarios la posibilidad de completar un cuestionario respondiendo a sus preguntas.	

Tabla 6. RC6- Completar cuestionario

4.1.2. Requisitos de restricción

ID	RR1
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Seguridad de la aplicación</i>	
Cualquier persona no puede acceder a la aplicación por lo que se ha de realizar un control de acceso al mismo. Para ello, por razones de seguridad, todos los usuarios que quieran acceder a la aplicación tendrán que hacerlo mediante la introducción del login de usuario y la contraseña.	

Tabla 7. RR1- Seguridad de la aplicación

ID	RR2
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Fácil manejo</i>	
La aplicación ha de ser de fácil manejo para el usuario, ya que éste puede que no tenga experiencia en el entorno informática. Debe constar de una de una interfaz intuitiva para el usuario.	
Además la aplicación se realizará de forma modular, para que ayude a la reutilización del código de la misma para posibles modificaciones futuras.	

Tabla 8. RR2- Fácil manejo



ID	RR3
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Paloma Fernández Ceballos
DESCRIPCIÓN: <i>Lenguaje en castellano</i>	
El lenguaje de la aplicación será el castellano.	

Tabla 9. RR3- Lenguaje en castellano

4.2. Casos de uso

En esta sección se exponen los casos de uso extraídos a partir de los requisitos de usuario.

En primer lugar se describen los actores implicados, a continuación se muestran las tablas que contienen la descripción de cada caso de uso, cada uno de ellos identificado unívocamente por un identificador. Justo a continuación de cada tabla, se mostrará el diagrama de actividad correspondiente con cada caso de uso y por último quedan plasmados los diagramas de casos de uso obtenidos.

4.2.1. Descripción de Actores

Realmente la aplicación tiene un único actor "Usuario" que es el que tiene mayor funcionalidad en la aplicación. Físicamente se corresponde con aquellas personas que han accedido al sistema mediante la introducción de una contraseña.

A su vez, un usuario puede ser de dos tipos:

- ✚ **Usuario "básico":** puede modificar su contraseña de acceso a la aplicación y puede también elegir un cuestionario y responderlo.
- ✚ **Usuario administrador:** hereda las funcionalidades del usuario básico, y además puede gestionar usuarios y sus datos personales, cuestionarios, secciones de los cuestionarios, preguntas de las secciones, y visualizar estadísticas relacionadas con las respuestas de los usuarios.

4.2.2. Descripción de Casos de Uso

A continuación se detallan los casos de uso en forma de tablas.

Nombre	<i>Loguearse</i>	Identificador	CU01
Actores	Usuario		
Objetivo	Acceder al menú principal de la aplicación. Relacionado con el requisito RR1.		
Precondiciones	Ninguna		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. El usuario introduce su login y password. 2. El usuario presiona el botón "Entrar". 3. La aplicación carga la página principal y el menú correspondiente dependiendo del tipo de usuario que sea, "básico" o "administrador". 		

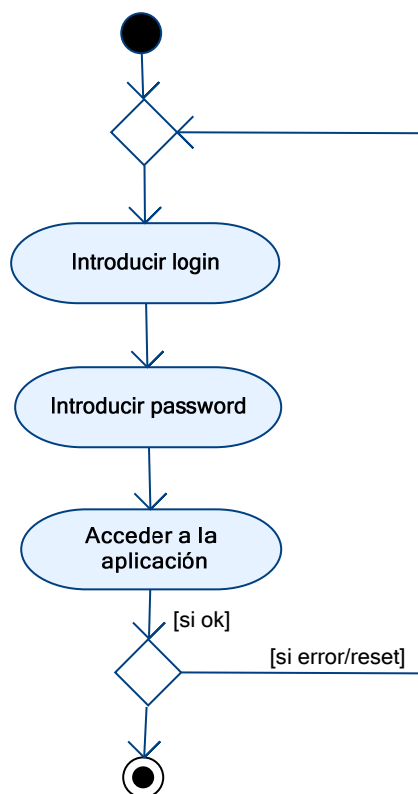


Figura 10. CU01- Loguearse

Nombre	<i>Cerrar sesión</i>	Identificador	CU02
Actores	Usuario		
Objetivo	Salir de la aplicación. Relacionado con el requisito RR1.		
Precondiciones	El usuario debe estar logueado en la aplicación previamente		
Postcondiciones	Ninguna		
Escenario Básico	1. El usuario hace <i>click</i> sobre el enlace "Cerrar sesión". 2. La sesión se cierra correctamente y la aplicación redirige al usuario a la pantalla de <i>login</i> .		

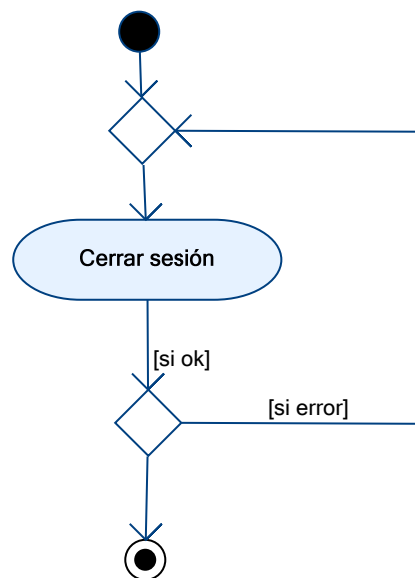


Figura 11. CU02- Cerrar sesión



Nombre	<i>Dar de alta a usuario</i>	Identificador	CU03
Actores	Usuario administrador		
Objetivo	Dar de alta a un usuario en la base de datos. Relacionado con el requisito RC1.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "USUARIOS", el usuario selecciona la opción "Dar de alta a usuario". 2. El usuario introduce login, nombre, apellidos, área de interés a la que pertenece, categoría de especialización y qué tipo de usuario es. 3. El usuario pulsa el botón de "Guardar cambios". 4. La aplicación muestra un mensaje de éxito de la operación. 		

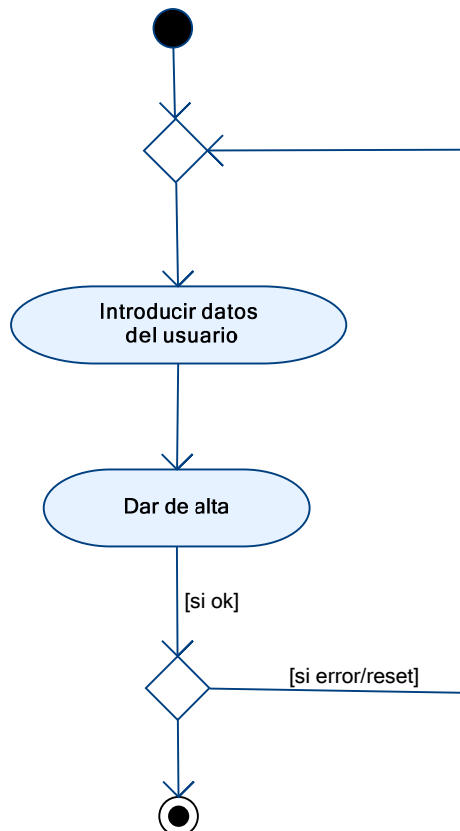


Figura 12. CU03- Dar de alta a usuario

Nombre	<i>Dar de baja a usuario</i>	Identificador	CU04
Actores	Usuario administrador		
Objetivo	Dar de baja a un usuario de la base de datos. Relacionado con el requisito RC1.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "USUARIOS", el usuario selecciona la opción "Dar de baja a usuario". 2. El usuario selecciona de la lista desplegable el login del usuario al que desea dar de baja. 3. Mediante la opción "Mostrar datos del usuario", el usuario verifica que es el usuario seleccionado el que quiere eliminar de la base de datos. 4. El usuario pulsa el botón de "Dar de baja". 5. La aplicación muestra un mensaje de éxito de la operación. 		

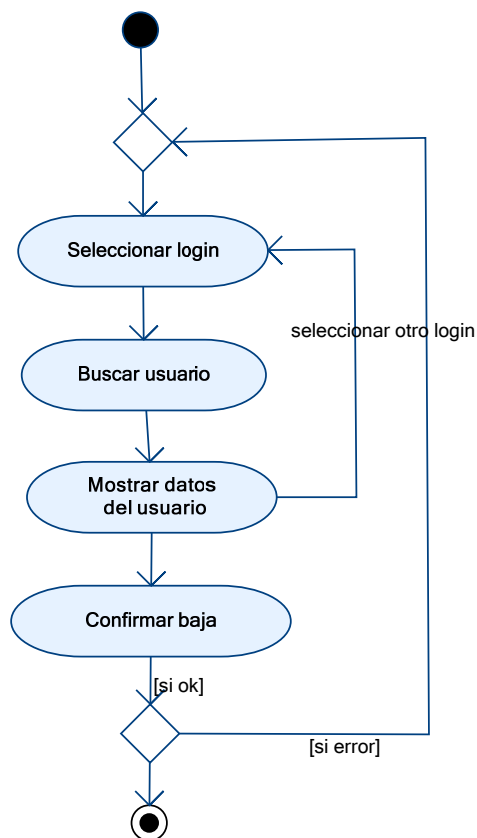


Figura 13. CU04- Dar de baja a usuario



Nombre	Modificar datos usuario	Identificador	CU05
Actores	Usuario administrador		
Objetivo	Modificar los datos personales de un usuario. Relacionado con el requisito RC1.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "USUARIOS", el usuario selecciona la opción "Modificar datos de un usuario". 2. El usuario selecciona de la lista desplegable el login del usuario al que desea modificar alguno de sus datos personales. 3. Mediante la opción "Mostrar datos del usuario", el usuario puede ver los datos personales del usuario que quiere modificar. 4. El usuario modifica el/los dato/s del usuario que procedan. Pueden ser el nombre, apellidos, área de interés a la que pertenece, categoría de especialización y qué tipo de usuario es. 5. El usuario pulsa el botón de "Guardar cambios". 6. La aplicación muestra un mensaje de éxito de la operación. 		

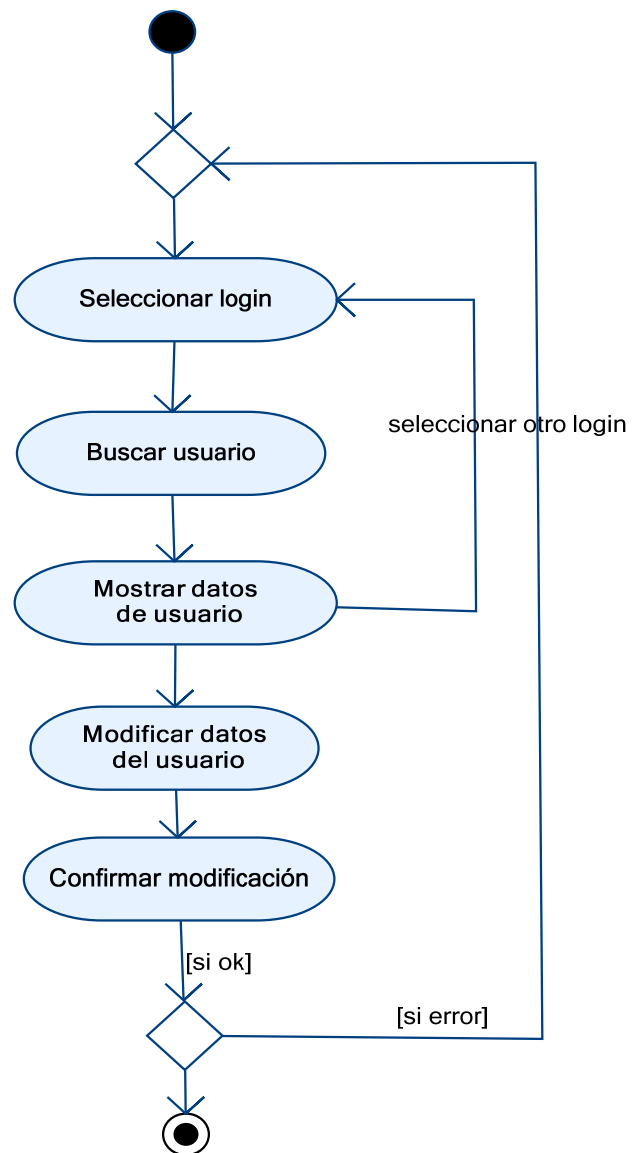


Figura 14. CU05- Modificar datos de usuario

Nombre	Consultar datos usuario	Identificador	CU06
Actores	Usuario administrador		
Objetivo	Consultar los datos personales de un usuario. Relacionado con el requisito RC1.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "USUARIOS", el usuario selecciona la opción "Consultar datos de un usuario". 2. El usuario selecciona de la lista desplegable el login del usuario del que quiere consultar sus datos. 3. Mediante la opción "Mostrar datos del usuario", el usuario puede ver los datos personales del usuario en cuestión. 		

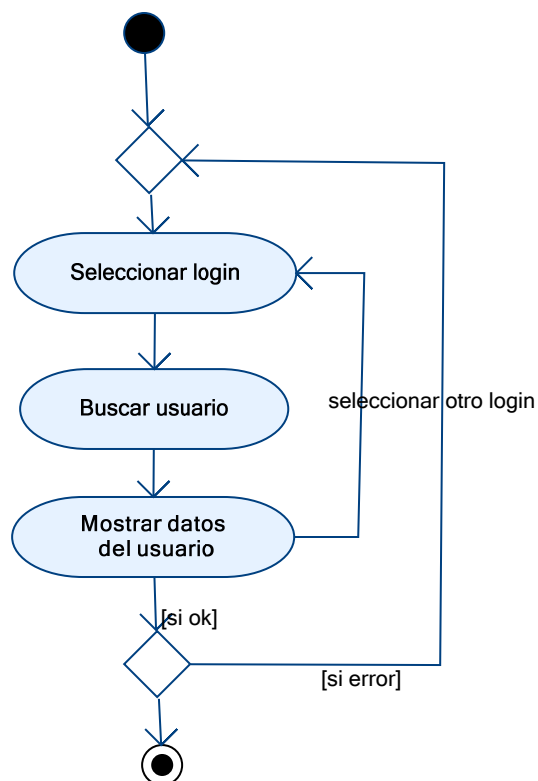


Figura 15. CU06- Consultar datos de usuario



Nombre	<i>Crear cuestionario</i>	Identificador	CU07
Actores	Usuario administrador		
Objetivo	Crear un cuestionario en la base de datos. Relacionado con el requisito RC2.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "CUESTIONARIOS", el usuario selecciona la opción "Crear cuestionario". 2. El usuario introduce el identificador, título y selecciona las secciones que va a contener. 3. El usuario pulsa el botón de "Guardar cambios". 4. La aplicación muestra un mensaje de éxito de la operación. 		

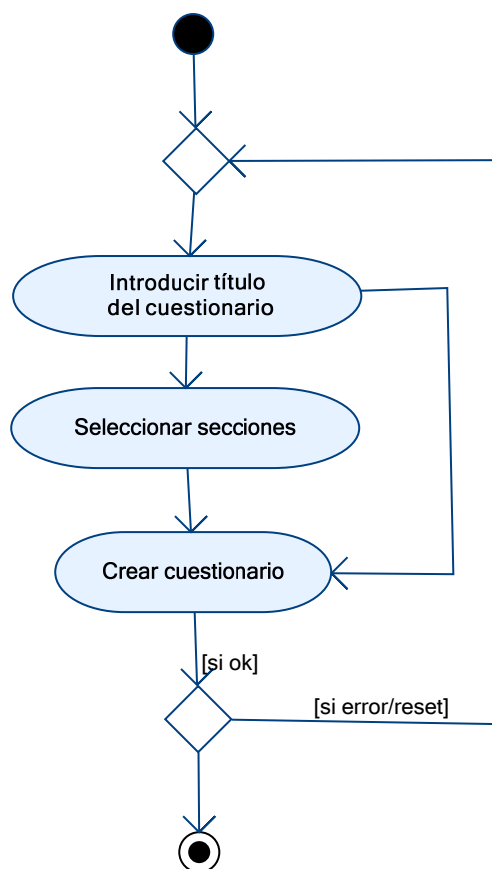


Figura 16. CU07- Crear cuestionario



Nombre	<i>Eliminar cuestionario</i>	Identificador	CU08
Actores	Usuario administrador		
Objetivo	Eliminar un cuestionario de la base de datos. Relacionado con el requisito RC2.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "CUESTIONARIOS", el usuario selecciona la opción "Eliminar cuestionario". 2. El usuario selecciona de la lista desplegable el título del cuestionario que desea eliminar. 3. Mediante la opción "Mostrar datos del cuestionario", el usuario verifica que es el cuestionario seleccionado el que quiere eliminar de la base de datos. 4. El usuario pulsa el botón de "Eliminar". 5. La aplicación muestra un mensaje de éxito de la operación. 		

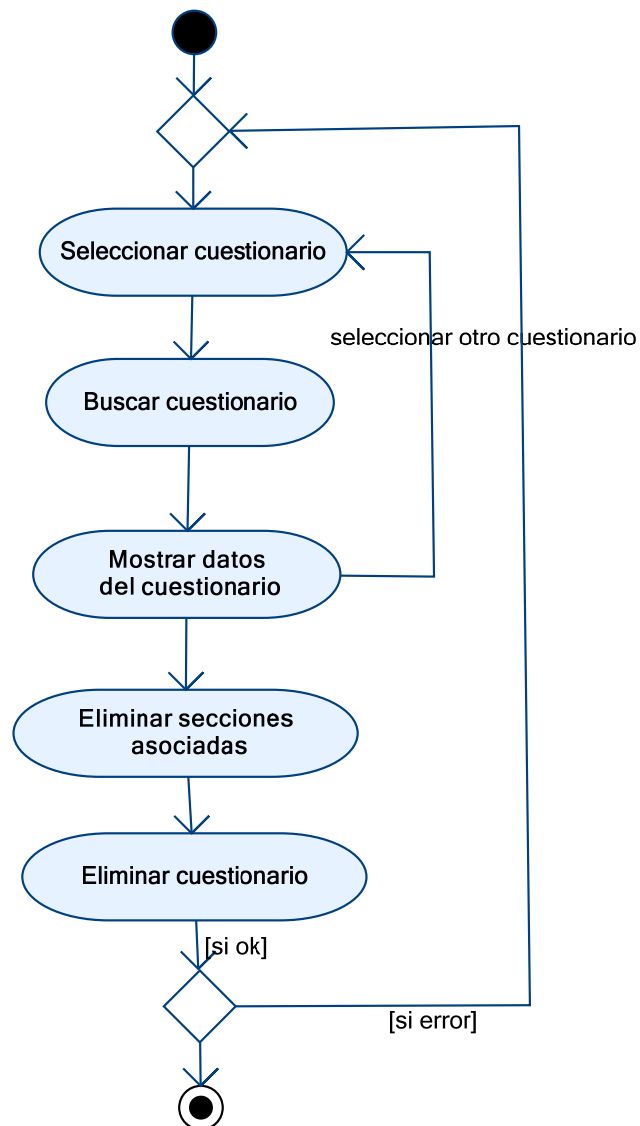
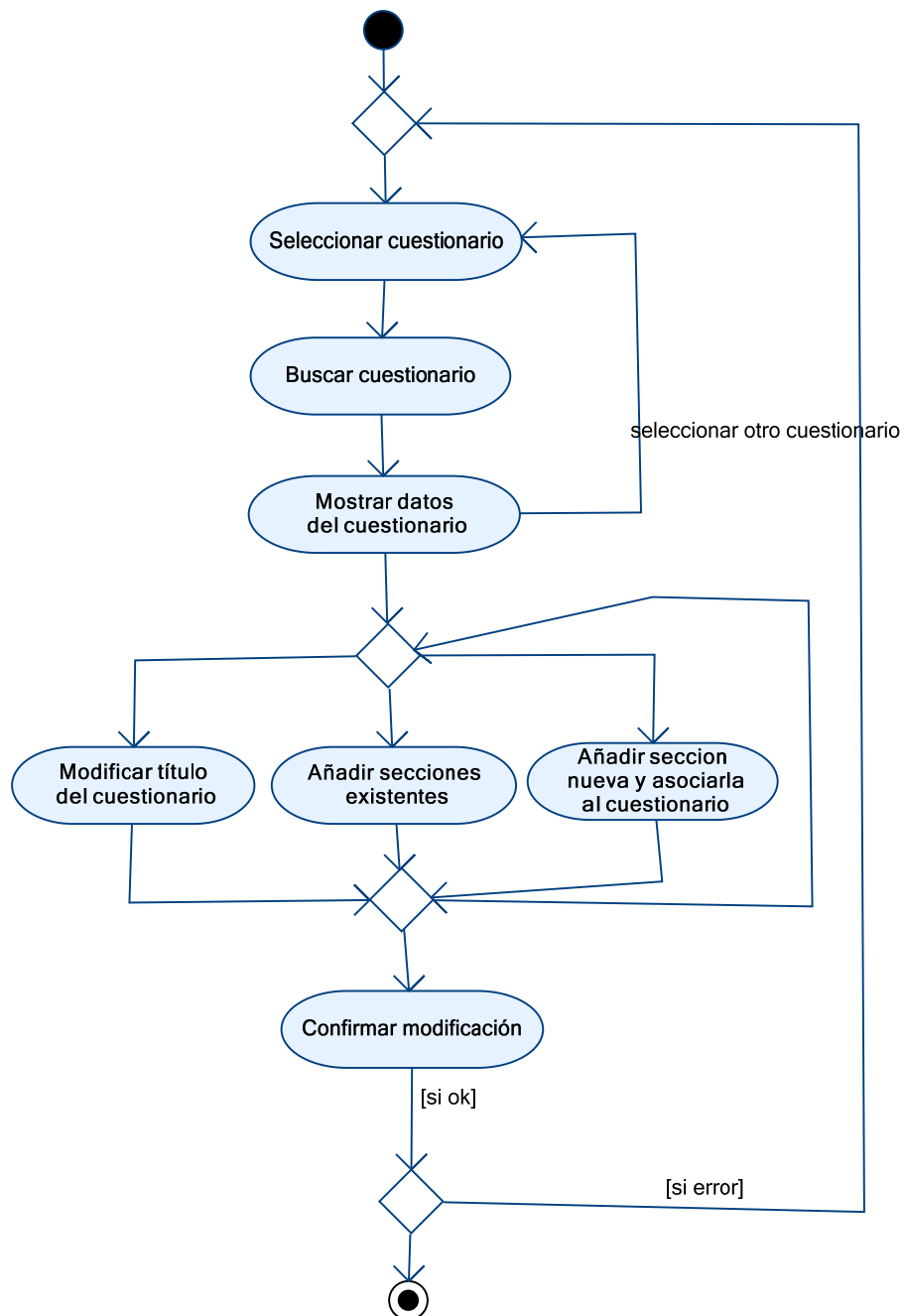


Figura 17. CU08- Eliminar cuestionario



Nombre	Modificar datos de un cuestionario	Identificador	CU09
Actores	Usuario administrador		
Objetivo	Modificar los datos de un cuestionario. Relacionado con el requisito RC2.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "CUESTIONARIOS", el usuario selecciona la opción "Modificar datos de un cuestionario". 2. El usuario selecciona de la lista desplegable el título del cuestionario al que desea modificar sus datos. 3. Mediante la opción "Mostrar datos del cuestionario", el usuario puede ver los datos del cuestionario que quiere modificar. 4. El usuario modifica el/los dato/s del cuestionario que procedan. Pueden ser el título, o las secciones. Podrá añadir tanto secciones nuevas como existentes, así como eliminar las que desee. 5. El usuario pulsa el botón de "Guardar cambios". 6. La aplicación muestra un mensaje de éxito de la operación. 		

**Figura 18. CU09- Modificar cuestionario**

Nombre	Consultar datos de un cuestionario	Identificador	CU10
Actores	Usuario administrador		
Objetivo	Consultar los datos de un cuestionario. Relacionado con el requisito RC2.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "CUESTIONARIOS", el usuario selecciona la opción "Consultar datos de un cuestionario". 2. El usuario selecciona de la lista desplegable el título del cuestionario que desea consultar. 3. Mediante la opción "Mostrar datos del cuestionario", el usuario puede ver los datos del cuestionario en cuestión. 		

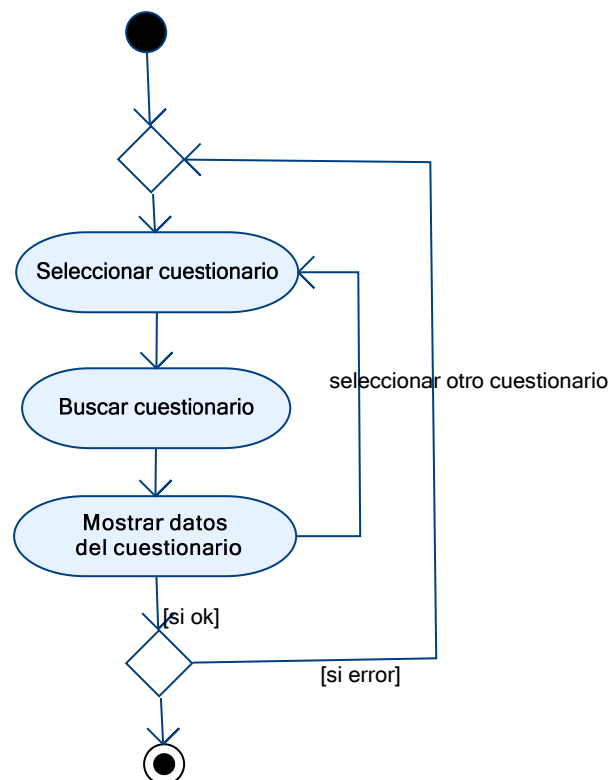


Figura 19. CU10- Consultar cuestionario



Nombre	Crear sección	Identificador	CU11
Actores	Usuario administrador		
Objetivo	Insertar una nueva sección en la base de datos. Relacionado con el requisito RC3.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "SECCIONES", el usuario selecciona la opción "Crear sección". 2. El usuario introduce el identificador, título y selecciona las preguntas que va a contener. 3. El usuario pulsa el botón de "Guardar cambios". 4. La aplicación muestra un mensaje de éxito de la operación. 		

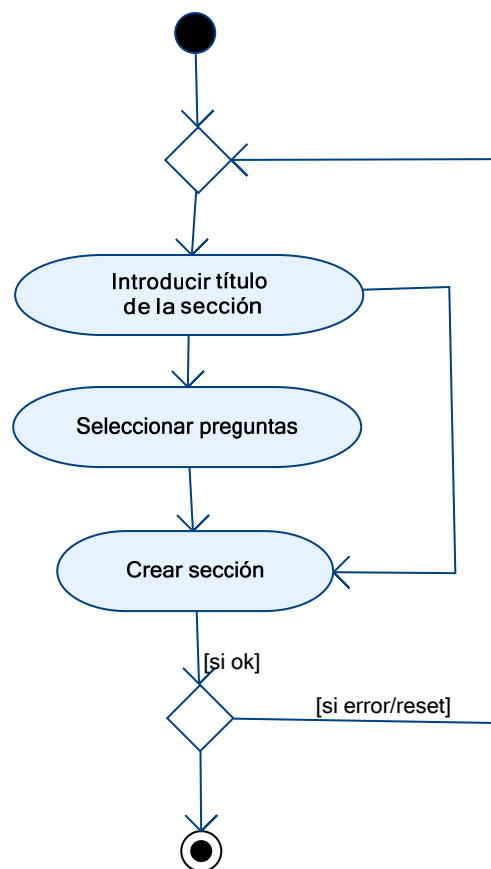


Figura 20. CU11- Crear sección



Nombre	<i>Eliminar sección</i>	Identificador	CU12
Actores	Usuario administrador		
Objetivo	Eliminar una sección de la base de datos. Relacionado con el requisito RC3.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "SECCIONES", el usuario selecciona la opción "Eliminar sección". 2. El usuario selecciona de la lista desplegable el título de la sección que desea eliminar. 3. Mediante la opción "Mostrar datos de la sección", el usuario verifica que es la sección seleccionada la que quiere eliminar de la base de datos. 4. El usuario pulsa el botón de "Eliminar". 5. La aplicación muestra un mensaje de éxito de la operación. 		

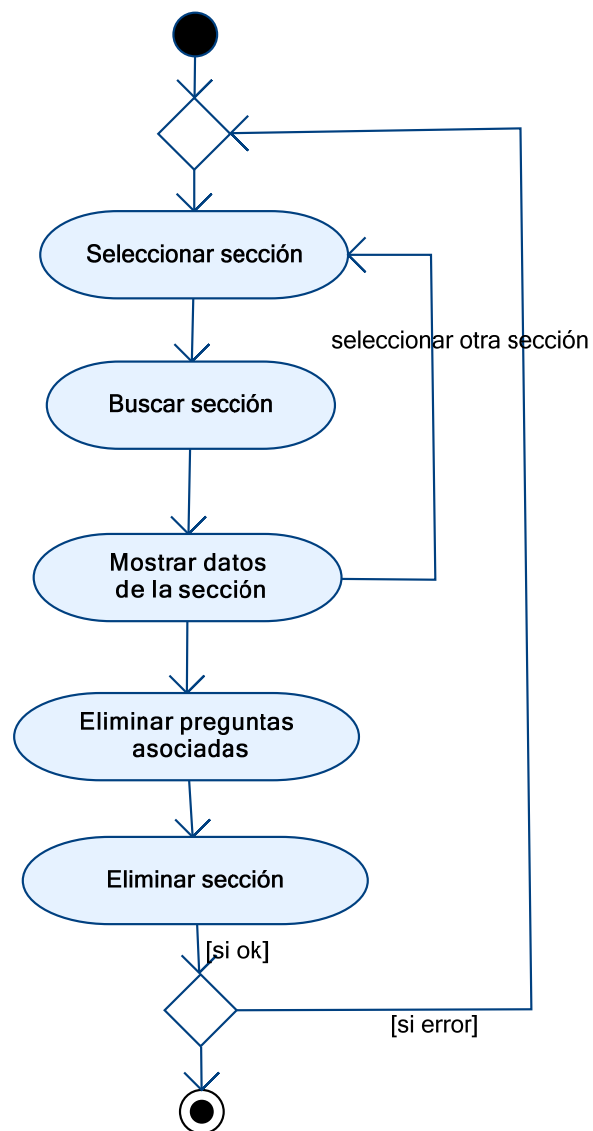


Figura 21. CU12- Eliminar sección



Nombre	Modificar datos de una sección	Identificador	CU13
Actores	Usuario administrador		
Objetivo	Modificar los datos de una sección. Relacionado con el requisito RC3.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "SECCIONES", el usuario selecciona la opción "Modificar datos de una sección". 2. El usuario selecciona de la lista desplegable el título de la sección a la que desea modificar sus datos. 3. Mediante la opción "Mostrar datos de la sección", el usuario puede ver los datos de la sección que quiere modificar. 4. El usuario modifica el/los dato/s de la sección que procedan. Pueden ser el título, o las preguntas. Podrá añadir tanto preguntas nuevas como existentes, así como eliminar las que desee. 5. El usuario pulsa el botón de "Guardar cambios". 6. La aplicación muestra un mensaje de éxito de la operación. 		

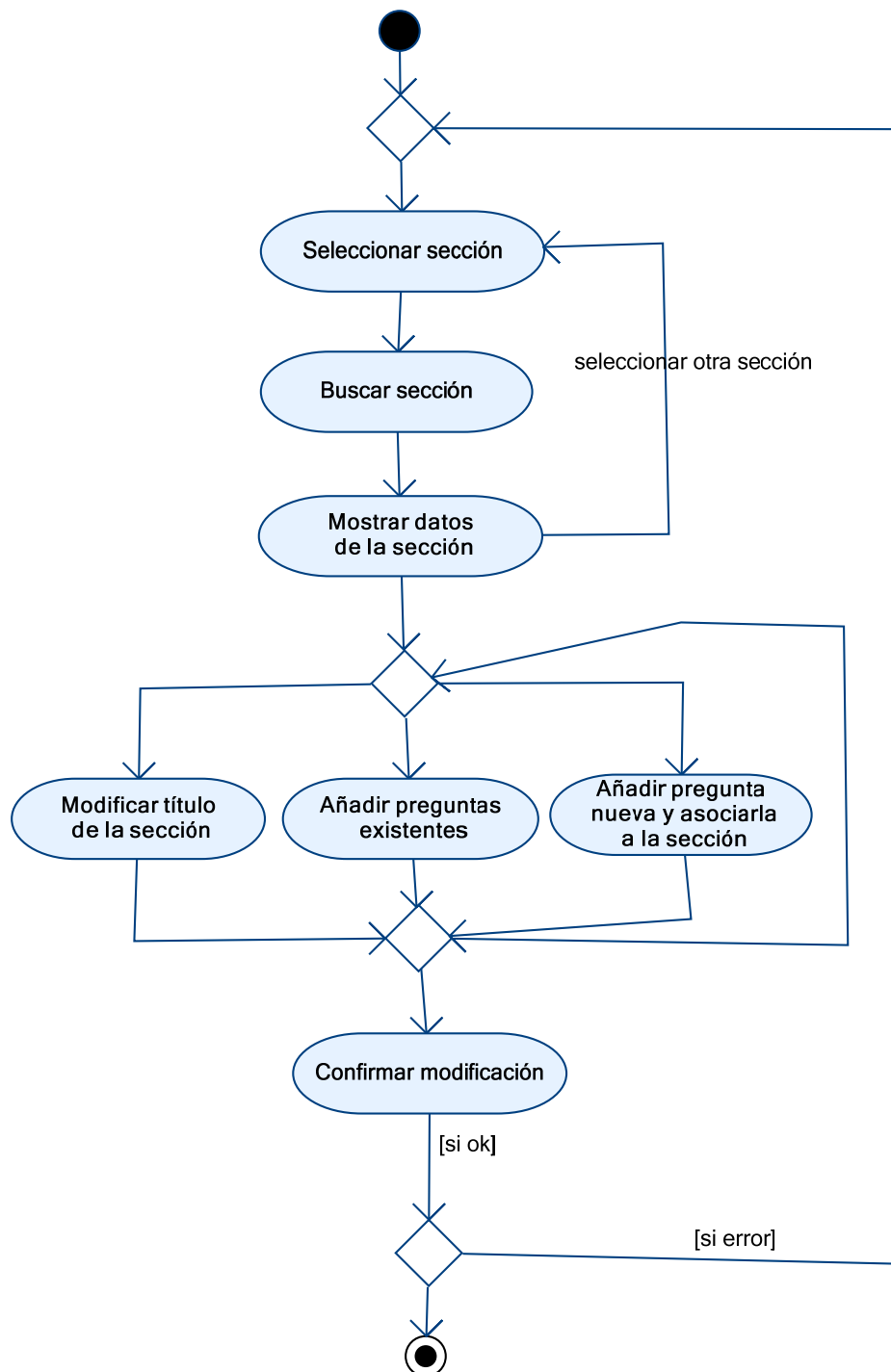


Figura 22. CU13- Modificar sección

Nombre	Consultar datos de una sección	Identificador	CU14
Actores	Usuario administrador		
Objetivo	Consultar los datos de una sección. Relacionado con el requisito RC3.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "SECCIONES", el usuario selecciona la opción "Consultar datos de una sección". 2. El usuario selecciona de la lista desplegable el título de la sección que desea consultar. 3. Mediante la opción "Mostrar datos de la sección", el usuario puede ver los datos de la sección en cuestión. 		

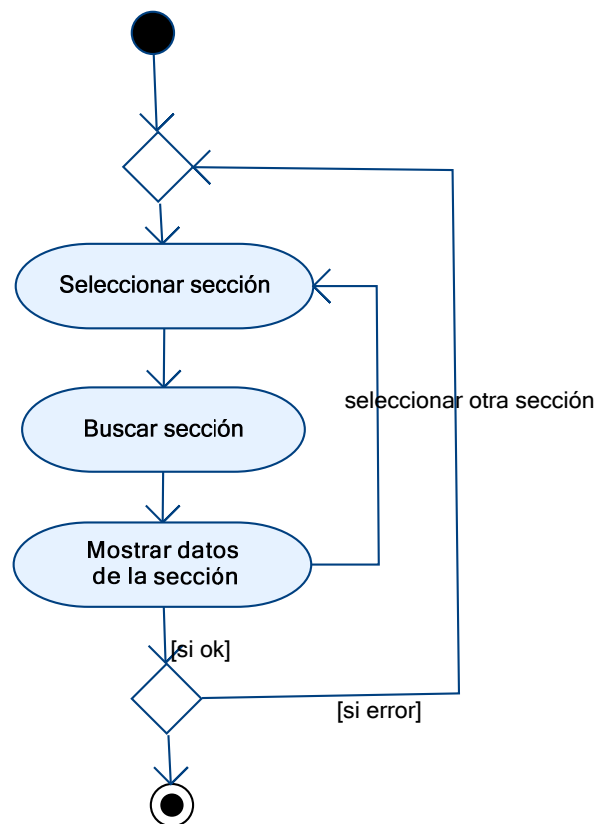


Figura 23. CU14- Consultar sección

Nombre	Crear pregunta	Identificador	CU15
Actores	Usuario administrador		
Objetivo	Insertar una nueva pregunta en la base de datos. Relacionado con el requisito RC4.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "PREGUNTAS", el usuario selecciona la opción "Crear pregunta". 2. El usuario introduce el identificador, título y tipo de respuestas que admitirá. 3. El usuario pulsa el botón de "Guardar cambios". 4. La aplicación muestra un mensaje de éxito de la operación. 		

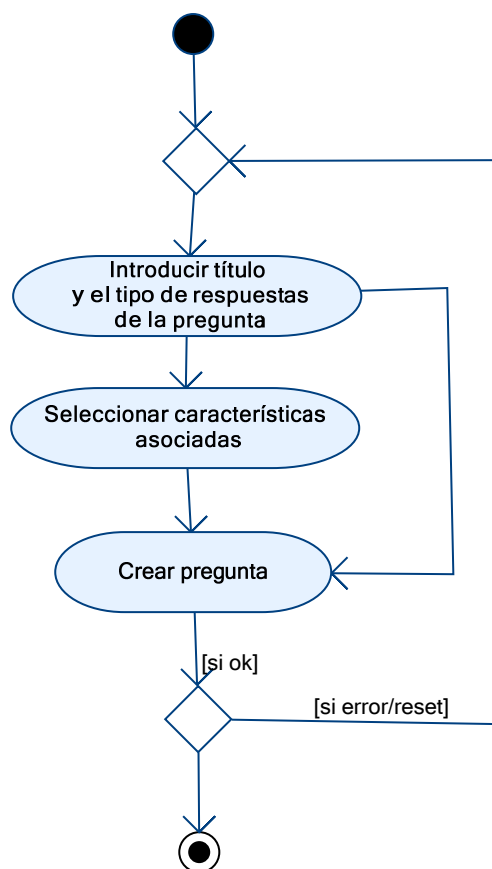


Figura 24. CU15- Crear pregunta

Nombre	<i>Eliminar pregunta</i>	Identificador	CU16
Actores	Usuario administrador		
Objetivo	Eliminar una pregunta de la base de datos. Relacionado con el requisito RC4.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "PREGUNTAS", el usuario selecciona la opción "Eliminar pregunta". 2. El usuario selecciona de la lista desplegable el título de la pregunta que desea eliminar. 3. Mediante la opción "Mostrar datos de la pregunta", el usuario verifica que es la pregunta seleccionada la que quiere eliminar de la base de datos. 4. El usuario pulsa el botón de "Eliminar". 5. La aplicación muestra un mensaje de éxito de la operación. 		

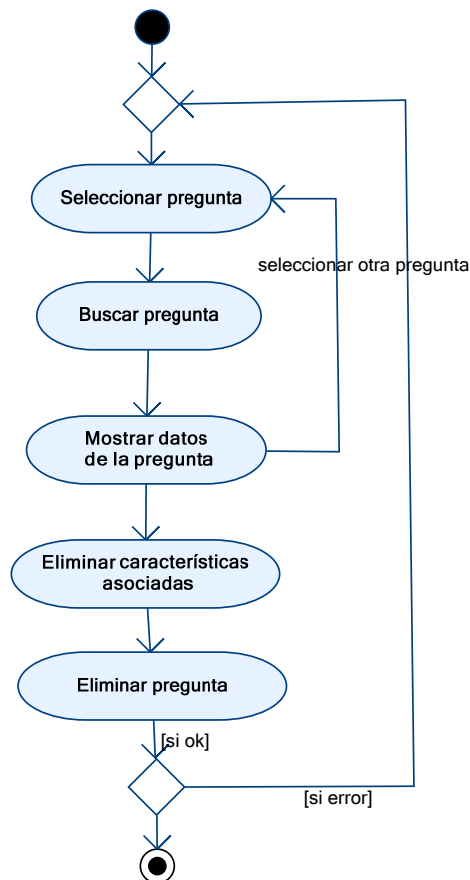


Figura 25. CU16- Eliminar pregunta



Nombre	Modificar datos de una pregunta	Identificador	CU17
Actores	Usuario administrador		
Objetivo	Modificar los datos de una pregunta. Relacionado con el requisito RC4.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "PREGUNTAS", el usuario selecciona la opción "Modificar datos de una pregunta". 2. El usuario selecciona de la lista desplegable el título de la pregunta a la que desea modificar sus datos. 3. Mediante la opción "Mostrar datos de la pregunta", el usuario puede ver los datos de la sección que quiere modificar. 4. El usuario modifica el/los dato/s de la pregunta que procedan. Pueden ser el título o el tipo de respuestas que admite. 5. El usuario pulsa el botón de "Guardar cambios". 6. La aplicación muestra un mensaje de éxito de la operación. 		

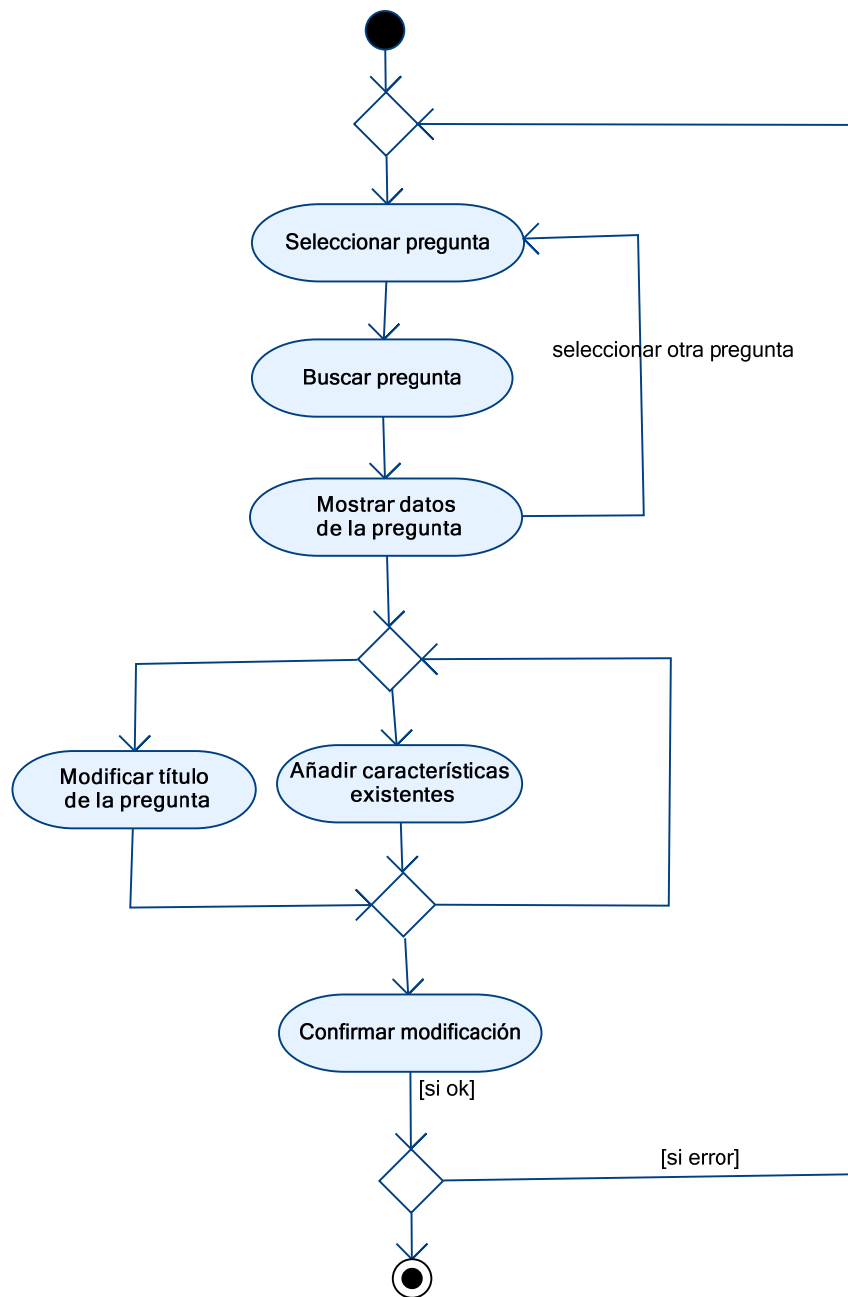


Figura 26. CU17- Modificar pregunta

Nombre	Consultar datos de una pregunta	Identificador	CU18
Actores	Usuario administrador		
Objetivo	Consultar los datos de una pregunta. Relacionado con el requisito RC4.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "PREGUNTAS", el usuario selecciona la opción "Consultar datos de una pregunta". 2. El usuario selecciona de la lista desplegable el título de la pregunta que desea consultar. 3. Mediante la opción "Mostrar datos de la pregunta", el usuario puede ver los datos de la pregunta en cuestión. 		

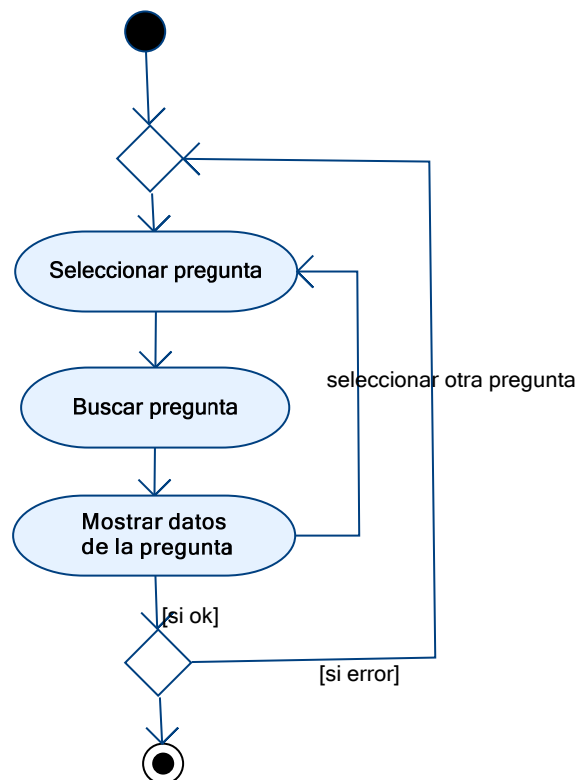


Figura 27. CU18- Consultar pregunta



Nombre	Completar cuestionario	Identificador	CU19
Actores	Usuario		
Objetivo	Responder un cuestionario. Relacionado con el requisito RC6.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "CUESTIONARIOS", el usuario selecciona la opción "Completar cuestionario". 2. El usuario selecciona de la lista desplegable el título del cuestionario que desea completar. 3. Mediante la opción "Responder cuestionario", el usuario puede ver los datos del cuestionario, de la primera sección que contiene y a su vez, de la primera pregunta de ésta. 4. El usuario rellena las puntuaciones de la primera pregunta. 5. El usuario presiona el botón siguiente pregunta. 6. Se vuelve al paso 4 siempre que el cuestionario siga teniendo preguntas. 7. El cuestionario no tiene más preguntas y la aplicación muestra un mensaje de agradecimiento por haberlo respondido. 		

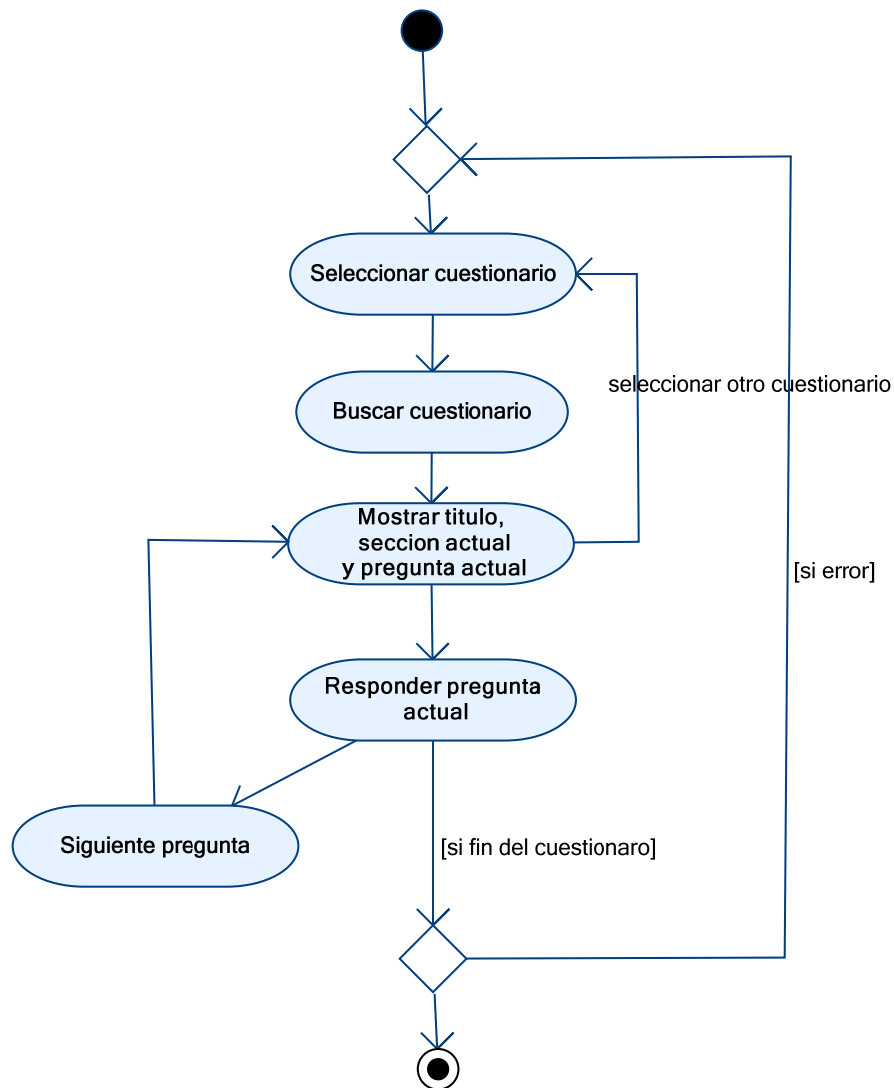


Figura 28. CU19- Completar cuestionario



Nombre	<i>Mostrar promedios de puntuaciones por pregunta y característica</i>	Identificador	CU20
Actores	Usuario administrador		
Objetivo	Mostrar el promedio de puntuación por pregunta y característica de un cuestionario. Relacionado con el requisito RC5.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "MOSTRAR RESULTADOS", el usuario selecciona la opción "Promedios". 2. El usuario selecciona de la lista desplegable el título del cuestionario del que desea consultar sus estadísticas. 3. Mediante la opción "Ver preguntas", el usuario puede ver las preguntas del cuestionario seleccionado. 4. Mediante la opción "Mostrar resultados", el usuario podrá ver los promedios de puntuaciones de la primera pregunta. 5. El usuario presiona el botón siguiente pregunta, para ver las estadísticas de la siguiente pregunta. 6. Se vuelve al paso 4 siempre que el cuestionario siga teniendo preguntas. 7. El cuestionario no tiene más preguntas y la aplicación muestra un mensaje de fin. 		

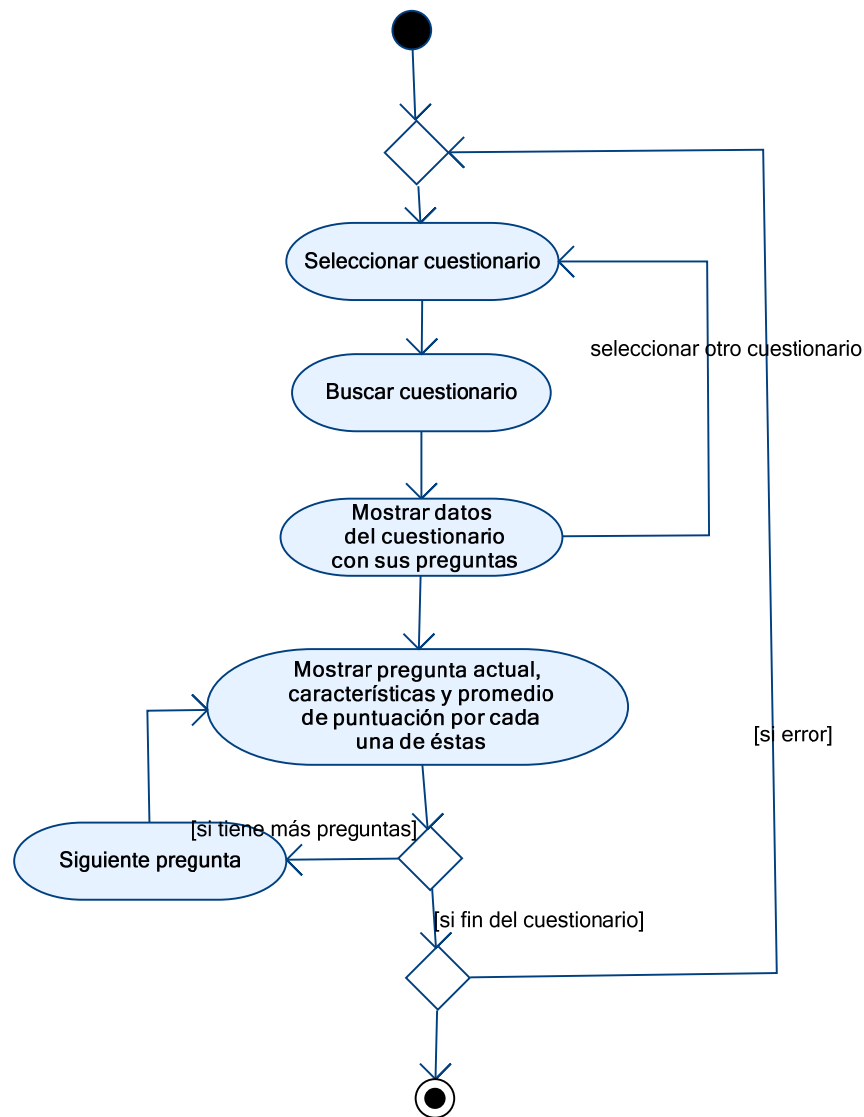


Figura 29. CU20- Mostrar promedios de puntuaciones por pregunta y característica



Nombre	<i>Mostrar porcentaje de usuarios que han respondido una de opción de cada pregunta y característica</i>	Identificador	CU21
Actores	Usuario administrador		
Objetivo	Mostrar el porcentaje de usuarios que han respondido cada opción por pregunta y característica de un cuestionario. Relacionado con el requisito RC5.		
Precondiciones	El usuario que vaya a realizar la acción debe estar previamente identificado como administrador en la aplicación.		
Postcondiciones	Ninguna		
Escenario Básico	<ol style="list-style-type: none"> 1. En el menú desplegable de la izquierda, en el apartado de "MOSTRAR RESULTADOS", el usuario selecciona la opción "Promedios". 2. El usuario selecciona de la lista desplegable el título del cuestionario del que desea consultar sus estadísticas. 3. Mediante la opción "Ver preguntas", el usuario puede ver las preguntas del cuestionario seleccionado. 4. Mediante la opción "Mostrar resultados", el usuario podrá ver los porcentajes de la primera pregunta. 5. El usuario presiona el botón siguiente pregunta, para ver las estadísticas de la siguiente pregunta. 6. Se vuelve al paso 4 siempre que el cuestionario siga teniendo preguntas. 7. El cuestionario no tiene más preguntas y la aplicación muestra un mensaje de fin. 		

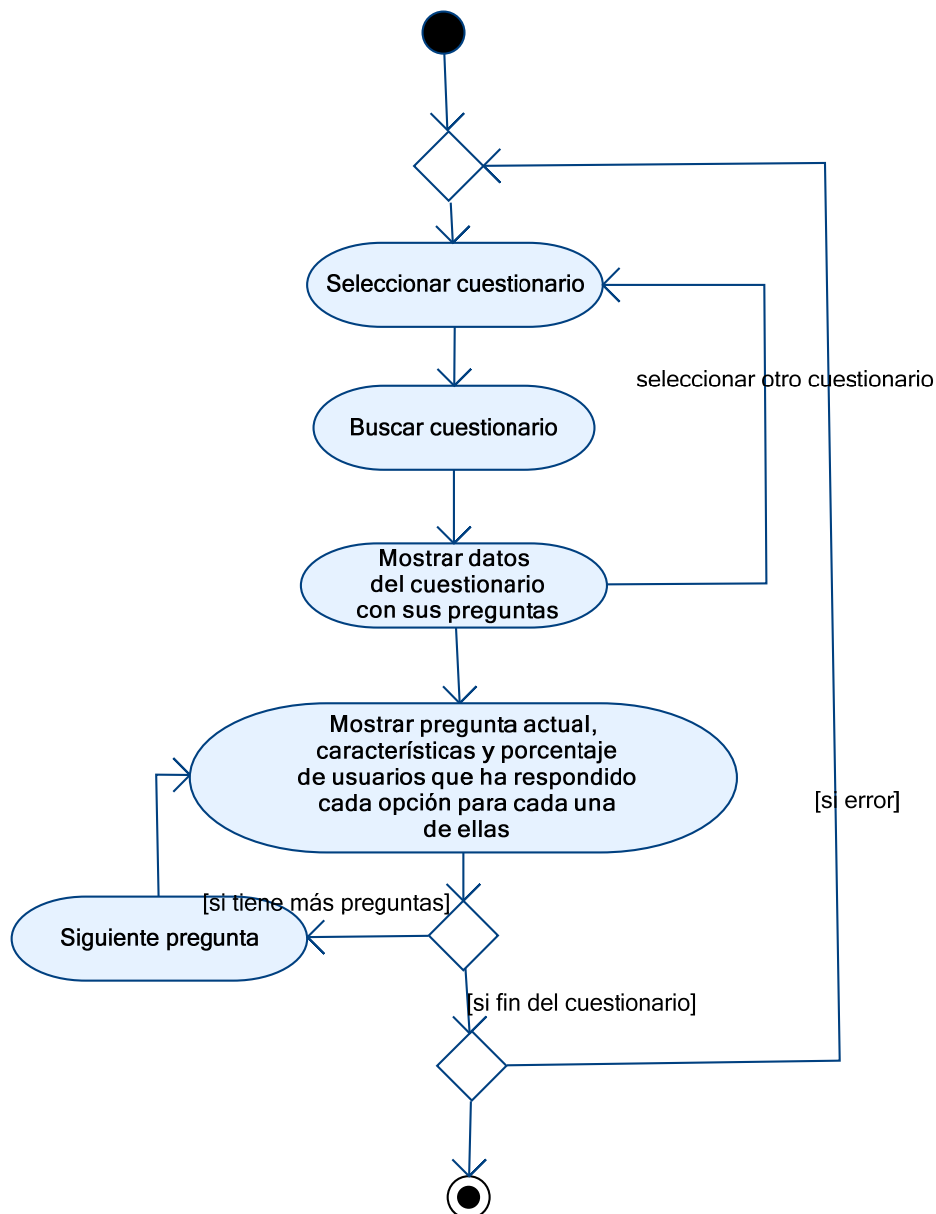


Figura 30. CU21- Mostrar porcentaje de usuarios que han respondido una de opción de cada pregunta y característica

4.2.3. Diagrama de Casos de Uso

Los diagramas de casos de uso se han dividido en tres escenarios:

- ✚ **Escenario 0:** Corresponde al login de usuario.
- ✚ **Escenario 1:** Acciones que puede realizar un usuario “básico”.
- ✚ **Escenario 2:** Abarca todas las funcionalidades de la aplicación, correspondientes al usuario administrador.

Escenario 0: *Login de usuario*

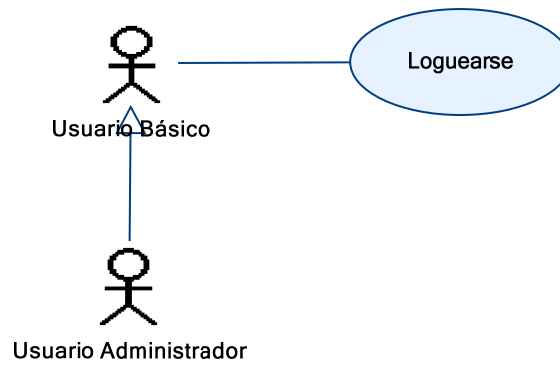


Figura 31. Escenario 0: Login de usuario

Escenario 1: *Funcionalidades para usuario básico*

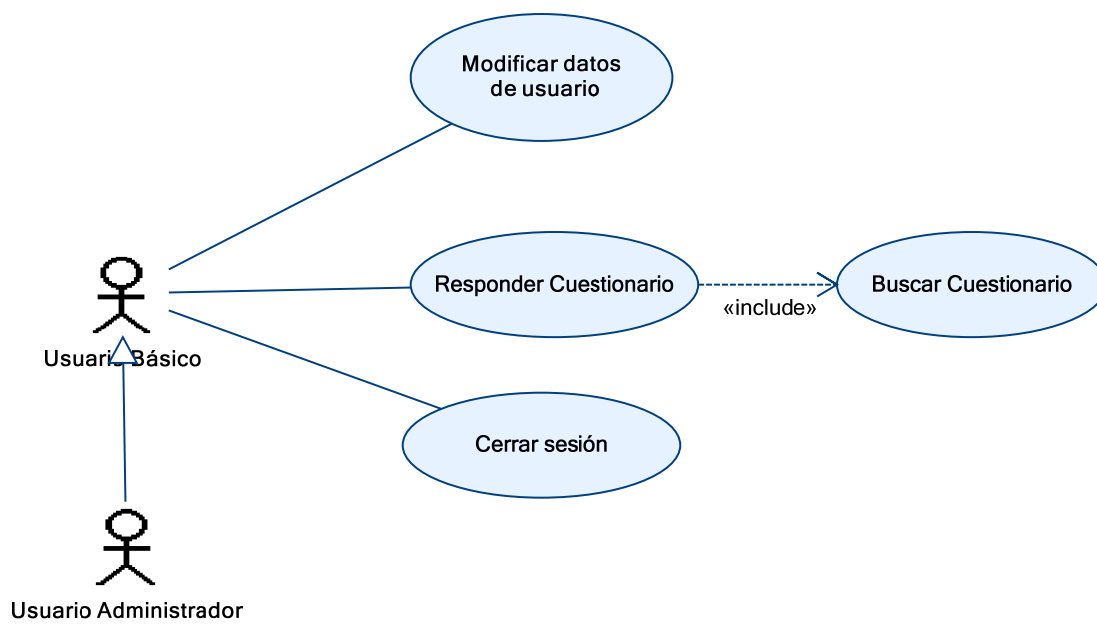
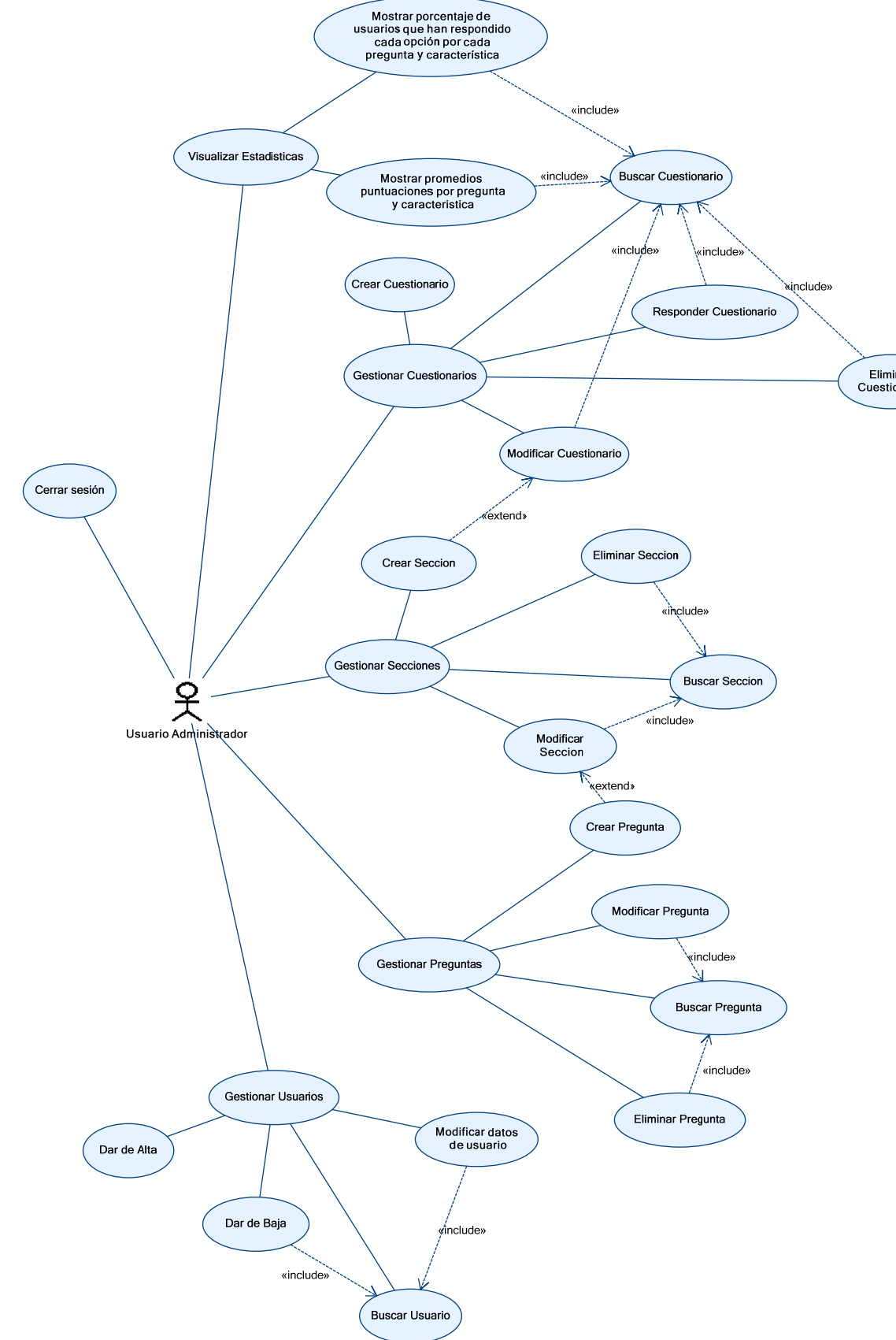


Figura 32. Escenario 1: Funcionalidades usuario básico



4.3. Modelo conceptual

El diagrama conceptual modela la vista estática del sistema y recoge conceptos del dominio de la aplicación. Se pretende dar una visión de los conceptos que maneja el sistema, los elementos que se utilizan para ello son clases y la relación entre ellas, atributos y métodos correspondientes.

Se muestra a continuación el diagrama de clases de obtenido en la fase de análisis:

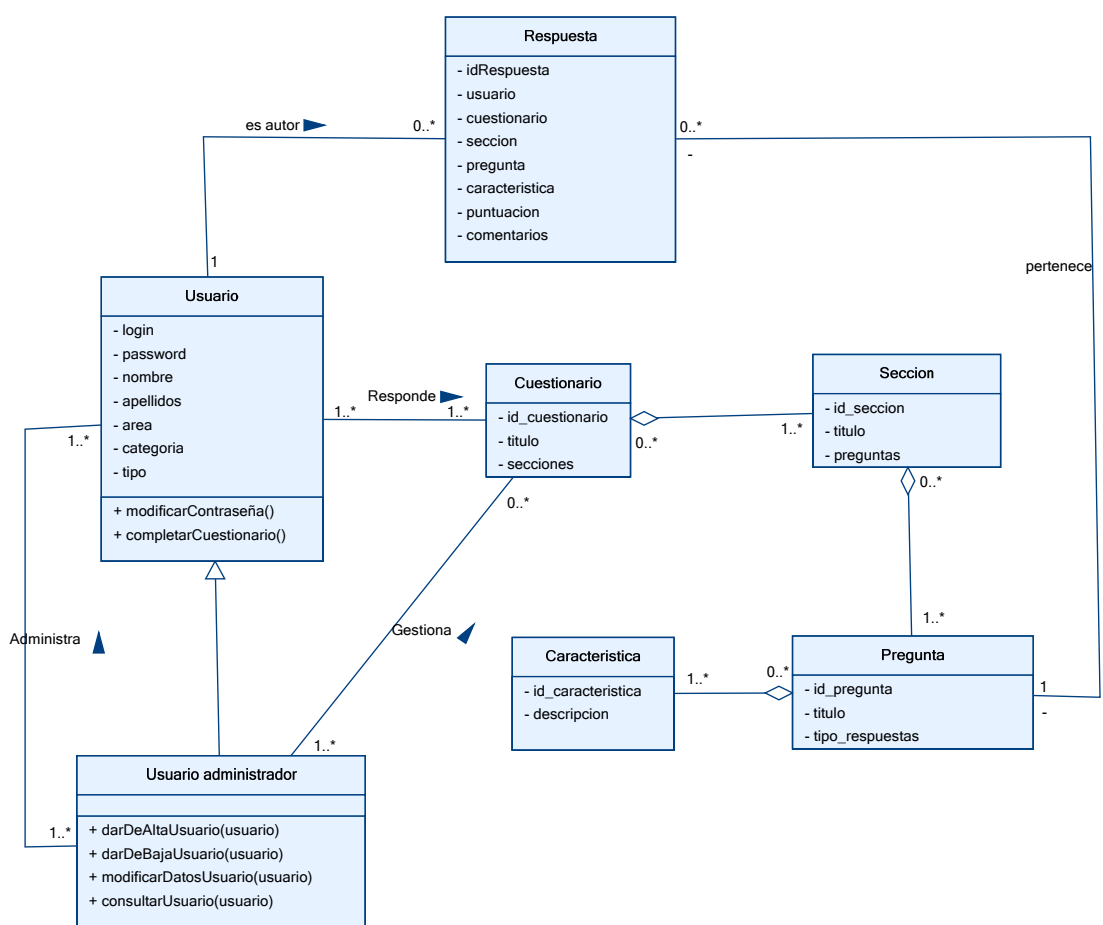


Figura 34. Diagrama de clases de análisis



5. DISEÑO DE LA APLICACIÓN

5.1. Requisitos de software

A partir de los requisitos definidos por el usuario, han sido extraídos los requisitos del software.

Se dividen en funcionales (derivados de los requisitos de capacidad) y no funcionales (vienen de los requisitos de restricción).

Dentro de los no funcionales se han extraído requisitos de interfaz, operacionales, de portabilidad y de documentación.

5.1.1. Requisitos funcionales

ID	RF01
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Dar de alta a usuario</i>	
La aplicación permitirá dar de alta a un usuario en la base de datos. Relacionado con el requisito RC1.	

Tabla 10. RF01- Dar de alta a usuario

ID	RF02
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Dar de baja a usuario</i>	
La aplicación permitirá eliminar a un usuario en la base de datos. Al dar de baja dicho usuario, se eliminarán todas sus respuestas. Relacionado con el requisito RC1.	

Tabla 12. RF02- Dar de baja a usuario

ID	RF03
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Consultar datos personales de un usuario</i>	
La aplicación permitirá consultar los datos personales de un usuario previamente seleccionado por su login. Dichos datos se corresponden con el nombre, apellidos, área de interés, categoría de especialización y tipo de usuario. Relacionado con el requisito RC1.	

Tabla 11. RF03- Consultar datos personales de usuario



ID	RF04
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Modificar datos personales de un usuario</i>	
La aplicación permitirá modificar los datos personales de un usuario previamente seleccionado por su login. Los datos que podrán modificar serán el nombre, apellidos, área de interés, categoría de especialización, contraseña y tipo de usuario. Relacionado con el requisito RC1.	

Tabla 13. RF04- Modificar datos personales de un usuario

ID	RF05
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Crear cuestionario</i>	
La aplicación permitirá crear un cuestionario que tendrá un título y un conjunto de secciones asociadas a él. Relacionado con el requisito RC2.	

Tabla 14. RF05- Crear cuestionario

ID	RF06
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Eliminar cuestionario</i>	
La aplicación permitirá eliminar un cuestionario de la base de datos. Con él se eliminarán todas sus secciones asociadas. Relacionado con el requisito RC2.	

Tabla 15. RF06- Eliminar cuestionario

ID	RF07
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Consultar cuestionario</i>	
La aplicación permitirá al usuario consultar las secciones de un cuestionario seleccionándolo previamente por su título. Asimismo, podrá consultar las preguntas de las secciones asociadas a él. Relacionado con el requisito RC2.	

Tabla 16. RF07- Consultar cuestionario



ID	RF08
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Modificar cuestionario</i>	
El usuario podrá modificar el título de un cuestionario, así como añadir o eliminar secciones al mismo. En el caso de añadir secciones, podrán ser existentes previamente en la base de datos, o desde esta funcionalidad se podrá crear una nueva sección que automáticamente se asociará a dicho cuestionario. Relacionado con el requisito RC2.	

Tabla 17. RF08- Modificar cuestionario

ID	RF09
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Crear sección</i>	
La aplicación permitirá crear una sección que tendrá un título y un conjunto de preguntas asociadas a ella. Relacionado con el requisito RC3.	

Tabla 18. RF09- Crear sección

ID	RF10
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Eliminar sección</i>	
La aplicación permitirá eliminar una sección de la base de datos. Con ella se eliminarán todas sus preguntas. Relacionado con el requisito RC3.	

Tabla 19. RF10- Eliminar sección

ID	RF11
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Consultar sección</i>	
La aplicación permitirá al usuario consultar las preguntas de una sección seleccionándola previamente por su título. Relacionado con el requisito RC3.	

Tabla 20. RF11- Consultar sección



ID	RF12
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Modificar sección</i>	
El usuario podrá modificar el título de una sección, así como añadir o eliminar preguntas a la misma. En caso de añadirla preguntas, podrán ser existentes previamente en la base de datos, o desde esta funcionalidad se podrá crear una nueva pregunta que automáticamente se asociará a dicha sección. Relacionado con el requisito RC3.	

Tabla 21. RF12- Modificar sección

ID	RF13
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Crear pregunta</i>	
La aplicación permitirá crear una pregunta que tendrá un título, un conjunto de características y un tipo de respuestas que determinarán las posibles puntuaciones que puede recibir dicha pregunta. Relacionado con el requisito RC4.	

Tabla 22. RF13- Crear pregunta

ID	RF14
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Eliminar pregunta</i>	
La aplicación permitirá eliminar una pregunta de la base de datos. Con ella se eliminarán todas las características asociadas a la misma. Relacionado con el requisito RC4.	

Tabla 23. RF14- Eliminar pregunta

ID	RF15
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Consultar pregunta</i>	
La aplicación permitirá al usuario consultar los de una pregunta seleccionándola previamente por su título. Dichos datos hacen referencia al tipo de respuestas y características asociadas a la misma. Relacionado con el requisito RC4.	

Tabla 24. RF15- Consultar pregunta



ID	RF16
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Modificar pregunta</i>	
El usuario podrá modificar el título y tipo de respuestas de una pregunta, así como añadir o eliminar características a la misma. Las características serán seleccionadas de una lista de características disponibles previamente guardadas en la base de datos. Relacionado con el requisito RC4.	

Tabla 25. RF16- Modificar pregunta

ID	RF17
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Completar cuestionario</i>	
La aplicación permitirá a todos los usuarios la posibilidad de rellenar un cuestionario contestando a sus preguntas. Relacionado con el requisito RC6.	

Tabla 26. RF17- Completar cuestionario

ID	RF18
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Visualizar promedios de puntuación por pregunta y característica</i>	
A partir de un cuestionario previamente seleccionado por el usuario, la aplicación permitirá a éste visualizar los promedios de las puntuaciones respondidas por pregunta y característica. Relacionado con el requisito RC5.	

Tabla 27. RF18- Visualizar promedios de puntuación por pregunta y característica

ID	RF19
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: <i>Visualizar porcentaje de usuarios que han respondido cada opción de una pregunta y característica</i>	
A partir de un cuestionario previamente seleccionado por el usuario, la aplicación permitirá a éste visualizar el porcentaje de usuarios que ha respondido cada opción (puntuación) para cada pregunta y característica. Relacionado con el requisito RC5.	

Tabla 28. RF19- Visualizar porcentaje de usuarios que han respondido cada opción de una pregunta y característica



ID	RF20
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: Acceso a la aplicación mediante login de usuario y contraseña	
El usuario accederá a la aplicación mediante la introducción de un login y una contraseña, lo que implica que para poder entrar en ella debe estar previamente registrado. Relacionado con el requisito RR1.	

Tabla 29. RF20- Acceso a la aplicación mediante login de usuario y contraseña

ID	RF21
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: Cerrar sesión	
Para salir de la aplicación correctamente, el usuario deberá cerrar sesión. De esta forma se borrarán todos los datos almacenados durante su sesión. Relacionado con el requisito RR1.	

Tabla 30. RF21- Cerrar sesión

5.1.2. Requisitos de interfaz

ID	RNF01
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles
DESCRIPCIÓN: Interfaz intuitiva y de fácil manejo	
La interfaz será sencilla e intuitiva para facilitar en la medida de lo posible la utilización de la misma por el usuario. Para ello, se colocará en el lado derecho de cada una de las vistas un desplegable con la ayuda disponible para cada funcionalidad. Relacionado con el requisito RR2.	

Tabla 31. RNF01- Interfaz intuitiva y de fácil manejo



ID	RNF02
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Paloma Fernández Ceballos
DESCRIPCIÓN: <i>Lengua de la interfaz: Castellano</i>	
Todo el texto contenido en las vistas estará escrito en castellano o términos técnicos comprensibles en dicha lengua. Relacionado con el requisito RR3.	

Tabla 32. RNF02- Lenguaje de la interfaz: Castellano

ID	RNF03
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Paloma Fernández Ceballos
DESCRIPCIÓN: <i>Interfaz optimizada para IE y Firefox</i>	
La interfaz estará optimizada para Internet Explorer 7.0 o superior y Mozilla Firefox 3 o superior.	

Tabla 33. RNF03- Interfaz optimizada para IE y Firefox

5.1.3. Requisitos operacionales

ID	RNF04
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Paloma Fernández Ceballos
DESCRIPCIÓN: <i>Las consultas a la base de datos no excederán los tres segundos.</i>	
La aplicación estará implementada de tal forma que las consultas a la base de datos no deberán exceder los tres segundos.	

Tabla 34. RNF04- Las consultas a la base de datos no excederán los tres segundos



5.1.4. Requisitos de portabilidad

ID	RNF05
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles y Paloma Fernández Ceballos
DESCRIPCIÓN: <i>Aplicación Web J2EE v5.0 + Struts 1.2.x + Hibernate 3.1</i>	
El sistema estará basado en una aplicación Web J2EE versión 5.0, con soporte para Struts 1.2.x así como Hibernate 3.1	

Tabla 35. RNF05- Aplicación Web J2EE + Struts + Hibernate

ID	RNF06
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles y Paloma Fernández Ceballos
DESCRIPCIÓN: <i>Sistema gestor de base de datos MySQL 5.0</i>	
El sistema gestor de base de datos a utilizar será MySQL 5.0	

Tabla 36. RNF06- Sistema gestor de base de datos MySQL 5.0

ID	RNF07
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles y Paloma Fernández Ceballos
DESCRIPCIÓN: <i>El servidor de aplicación será JBoss 1.4.x</i>	
El contenedor web utilizado será JBoss en su versión 1.4.x	

Tabla 37. RNF07- Servidor de aplicaciones JBoss 1.4.x

5.1.5. Requisitos de documentación

ID	RNF08
NECESIDAD	Esencial
ESTABILIDAD	Estable
FUENTE	Carmen Luisa Padrón Nápoles y Paloma Fernández Ceballos
DESCRIPCIÓN: <i>Entrega de documentación adjunta</i>	
Entrega de cd al usuario con la aplicación y una memoria que contendrá el análisis y diseño de la aplicación así como un manual de usuario.	







Tabla 38. RNF08- Entrega de documentación adjunta



5.2. Diseño de la capa de persistencia

Para el diseño de la capa de persistencia se han tenido en cuenta las entidades lógicas necesarias para el correcto almacenamiento de la información.

A continuación se describen las principales:

-  **Usuario** → Almacena la información asociada a los usuarios del sistema. Un usuario vendrá definido por un login, nombre, apellidos, área de interés, categoría de especialización y tipo de usuario (básico o administrador). Se identifica unívocamente por el login.
-  **Cuestionario** → Almacena la información asociada con un cuestionario. Un cuestionario vendrá definido por un identificador (por el que se identifica) y un título. Además un cuestionario estará formado por secciones, esta relación entre sección y cuestionario se guardará en la tabla SECCIONES_CUESTIONARIO.
-  **Sección** → Entidad lógica que almacena la información asociada con una sección. La definen un identificador (clave) y un título. Una sección estará formada por un conjunto de preguntas que estarán registradas en la tabla PREGUNTAS_SECCION.
-  **Pregunta** → Almacena la información correspondiente a una pregunta, que vendrá dada por un identificador con el cual se identifica unívocamente en la base de datos, un título y un tipo de respuestas.
El tipo de respuestas, mediante la relación entre TIPO_RESPUESTAS y VALORES_TIPO_RESPUESTAS, se encarga de asociar a la pregunta una serie de puntuaciones permitidas a la hora de responderla.
A su vez, una pregunta estará formada por características que se guardarán en la tabla CARACTERISTICAS_PREGUNTA.
-  **Característica** → Entidad lógica que almacena la información asociada a una característica, en este caso, un identificador y una descripción de la característica.
-  **Respuesta** → Entidad lógica que almacena la información asociada a una respuesta de un usuario.
Cada respuesta vendrá identificada unívocamente por un identificador que se generará de forma aleatoria al procesar la respuesta.
Una respuesta irá asociada a una pregunta, y a su vez, cada pregunta tendrá una respuesta por característica. Dicha pregunta



pertenecerá a una sección y esa sección formará parte de un cuestionario dado.

Además una respuesta deberá tener una puntuación y unos comentarios que será un campo opcional a la hora de rellenar por el usuario.

Se tendrán tres entidades adicionales para guardar las áreas de interés, las categorías de especialización y el tipo de usuario. Las tres relacionadas con la entidad Usuario y que tendrán como atributos, un identificador y una descripción.

Para el almacenamiento persistente de los objetos y estados de los mismos, en este caso implementado con una base de datos relacional, se ha utilizado **Hibernate**.

La Figura 35 muestra el grafo relacional que describe con detalle el diseño de la base de datos:

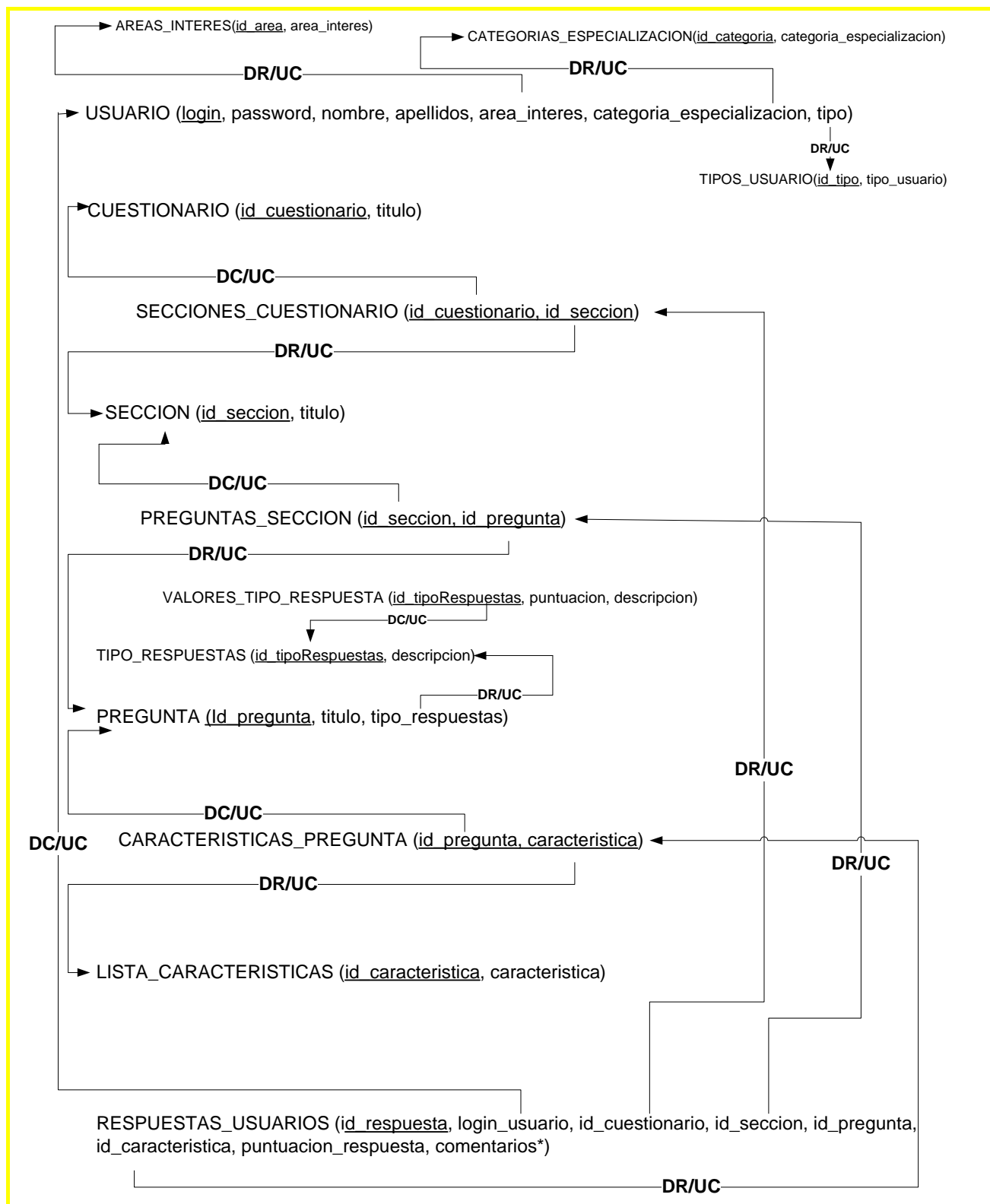


Figura 35. Diagrama relacional de la base de datos



5.3. Diseño de la capa de lógica de negocio

El diseño de la lógica de negocio de la aplicación está basado en el diagrama de clases que se muestra en la Figura 36.

Un modelo o diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargarán del funcionamiento y la relación entre uno y otro.

El modelo de clases de la aplicación obtenido en la fase de diseño consta de doce clases que representan las entidades lógicas que componen la capa de negocio:

- ✚ **UsuarioVO**: Clase que gestiona los usuarios. Contiene los atributos propios de un usuario y define las funcionalidades necesarias para validar un usuario, saber si tiene respondido algún cuestionario, darle de alta o de baja en la base de datos, consultar sus datos personales así como modificarlos, contiene también un método que lista todos los usuarios así como los *get* y *set* para sus atributos.
- ✚ **TiposUsuarioVO**: Clase que gestiona los tipos de usuario. En principio habrá dos tipos de usuarios: básico y administrador. Esta clase tiene las funcionalidades necesarias para la gestión de sus atributos (métodos *get* y *set*). Además contiene un método que devuelve una lista con los tipos de usuario registrados en la base de datos y otro método que devuelve la descripción del tipo de usuario a partir de su código.
- ✚ **AreasInteresVO**: Clase que gestiona las áreas de interés a las que puede pertenecer un usuario. Esta clase tiene las funcionalidades necesarias para la gestión de sus atributos (métodos *get* y *set*). Además contiene un método que devuelve una lista con las áreas de interés registradas en la base de datos y otro método que devuelve la descripción del área de interés a partir de su código.
- ✚ **CategoriasEspecializacionVO**: Clase que gestiona las categorías de especialización a las que puede pertenecer un usuario. Esta clase tiene las funcionalidades necesarias para la gestión de sus atributos (métodos *get* y *set*). Además contiene un método que devuelve una



lista con las categorías de especialización registradas en la base de datos y otro método que devuelve la descripción de una categoría a partir de su código.

- ✚ **CuestionarioVO**: Clase que se encarga de administrar la información asociada a los cuestionarios. Contiene los atributos propios de un cuestionario y define las funcionalidades necesarias para crear un cuestionario, eliminarlo, consultar sus datos o modificarlo, obtener una lista con las secciones asociadas a un determinado cuestionario, u obtener un listado con los cuestionarios disponibles en la base de datos.
- ✚ **SeccionVO**: Clase que se encarga de administrar la información asociada a las secciones. Contiene los atributos propios de una sección y define las funcionalidades necesarias para crear una sección, eliminarla, consultar sus datos o modificarla, obtener una lista con las preguntas asociadas a una determinada sección, u obtener un listado con las secciones disponibles en la base de datos.
- ✚ **PreguntaVO**: Clase que se encarga de administrar la información asociada a las preguntas. Contiene los atributos propios de una pregunta y define las funcionalidades necesarias para crear una pregunta, eliminarla, consultar sus datos o modificarla, obtener una lista con las características asociadas a una determinada pregunta, u obtener un listado con las preguntas disponibles en la base de datos.
- ✚ **CaracteristicaVO**: Clase que se encarga de administrar la información asociada a las características de las preguntas. Contiene los atributos propios de una característica y define las funcionalidades necesarias para consultar los datos de una determinada característica u obtener un listado de características.
- ✚ **TipoRespuestasVO**: Clase que se encarga de administrar la información asociada al tipo de respuesta correspondiente con una determinada pregunta. Contiene los atributos propios de un tipo de respuesta, estos son, un identificador y una descripción. Además contiene un método que devuelve una lista con los tipos de respuestas disponibles en la base de datos y otro método que devuelve la descripción del tipo de respuesta a partir de su código.
- ✚ **ValoresTipoRespuestasVO**: Clase que almacena los diferentes valores que puede tomar un determinado tipo de respuesta. Los atributos que la forman son un objeto de tipo TipoRespuestasVO, una puntuación y una descripción. Define las funcionalidades necesarias para obtener las puntuaciones permitidas a partir de un tipo de



respuesta así como un método que a partir de una tipo de respuesta y una puntuación devuelve la descripción de esa puntuación, por ejemplo, *nada*, *poco*, *mucho*, etc.

- ✚ **PuntuacionesPermitidasPreguntaVO:** Clase “auxiliar” que almacena un objeto de tipo `PreguntaVO` y otro de tipo `ValoresTipoRespuestasVO` con el fin de en una misma clase tener una pregunta y las puntuaciones que permite a la hora de responderla. Contiene los constructores básicos así como los métodos *get* y *set* para los atributos.
- ✚ **RespuestasUsuarioVO:** Clase que gestiona las respuestas de los usuarios. Contiene los atributos propios de una respuesta y define las funcionalidades necesarias para insertar una respuesta en la base de datos y eliminar las respuestas de un determinado usuario.

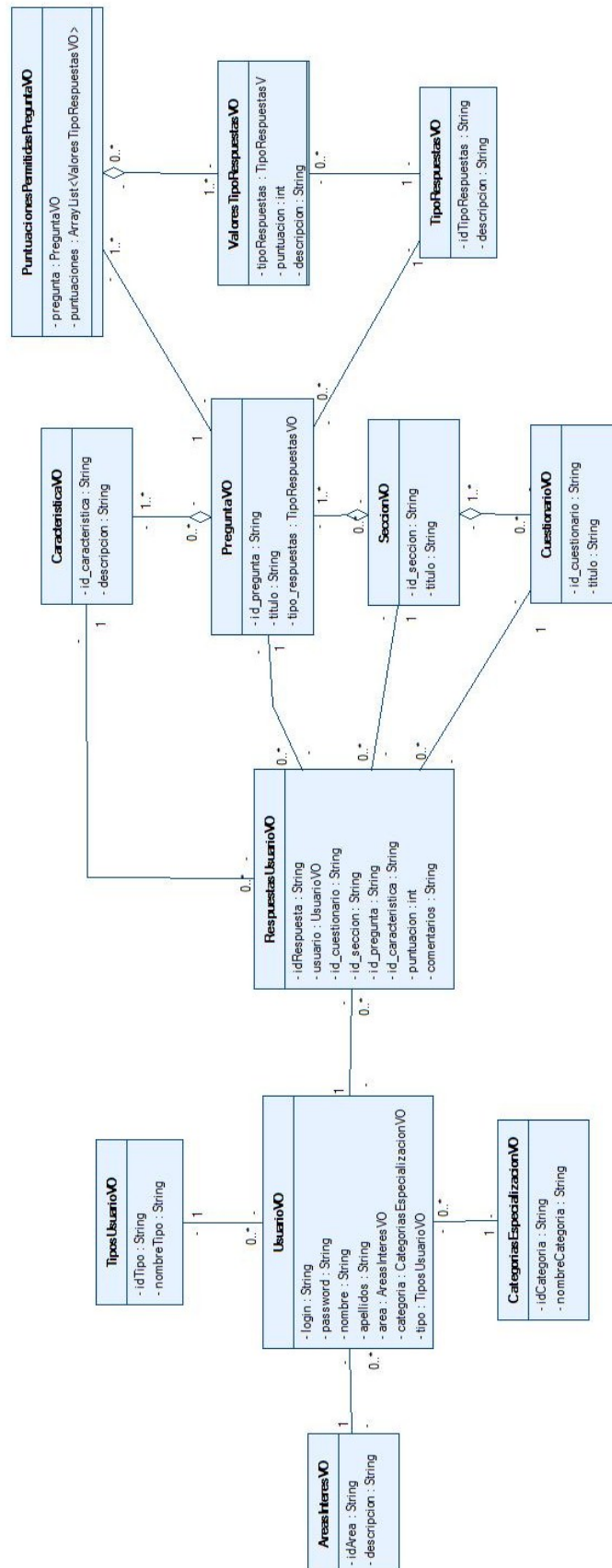


Figura 36. Diagrama de clases detallado

Hasta este punto queda modelada la parte estática del sistema, ahora es necesario modelar los aspectos dinámicos del mismo. Para ello se hará uso de los diagramas de secuencia.

Un **diagrama de secuencia** permite mostrar las interacciones entre objetos en el transcurso del tiempo; esto es, permite representar los objetos que participan en un escenario de un caso de uso: los eventos generados por actores externos, su orden y los eventos internos del sistema.

En un diagrama de secuencias se muestran dos dimensiones: la vertical que representa el tiempo, y la horizontal en la que se representan los distintos objetos.

Debido a la complejidad de los diagramas de secuencia obtenidos, se han dividido, toda la secuencia del sistema se ha dividido en cinco diagramas:

- **Secuencia del sistema para un usuario básico**

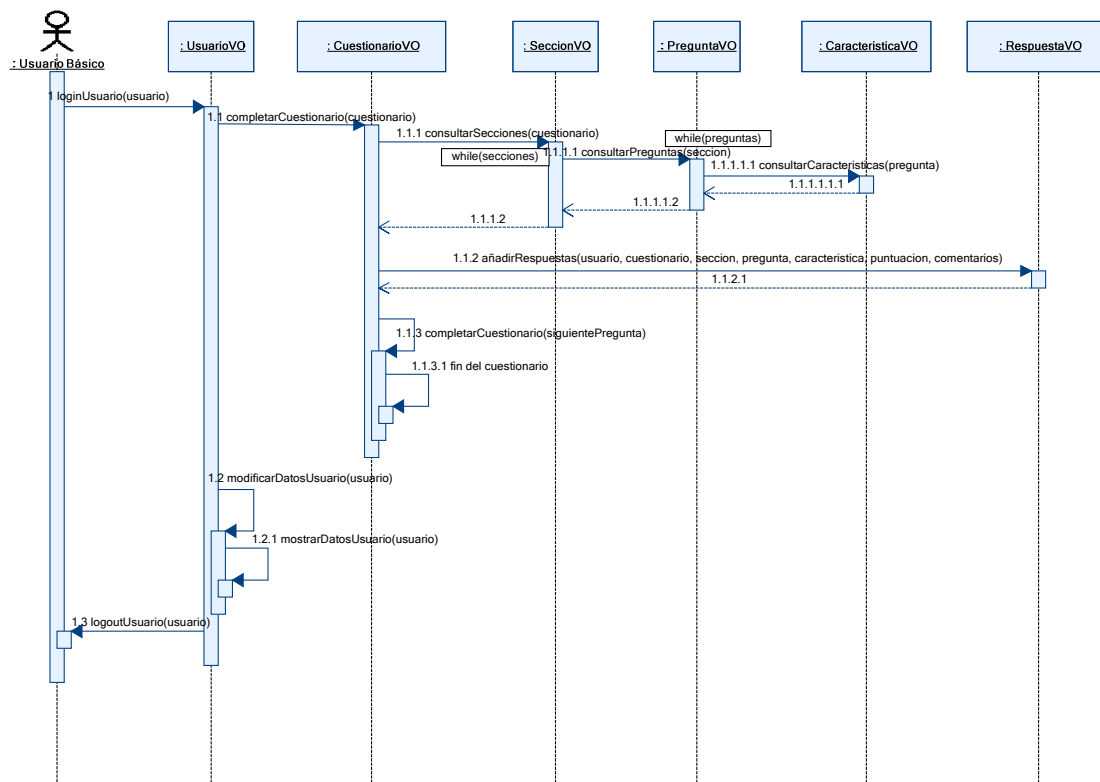


Figura 37. Diagrama de secuencia- Usuario básico



- **Gestión de usuarios**

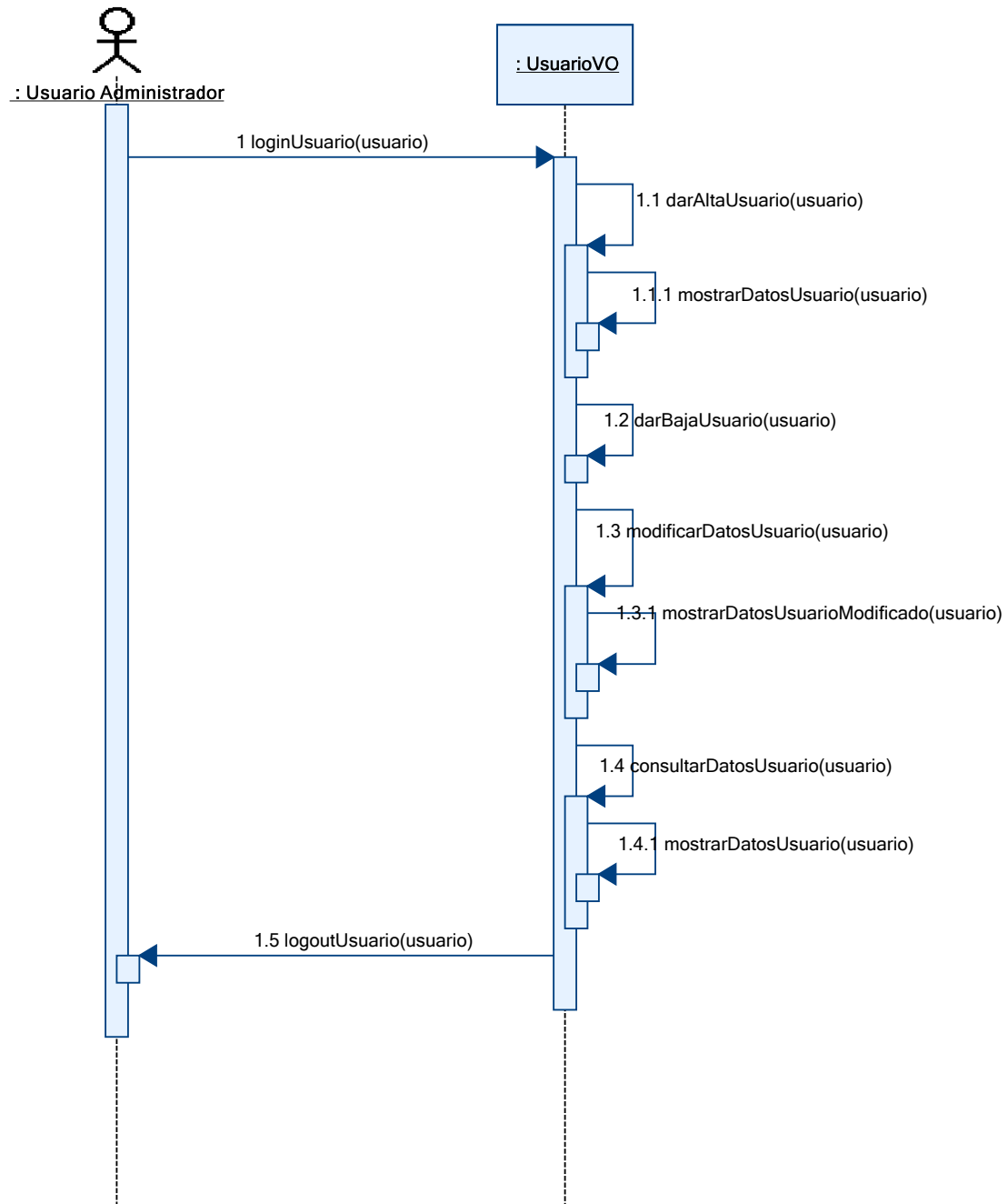


Figura 38. Diagrama de secuencia- Gestión de usuarios

• Gestión de cuestionarios

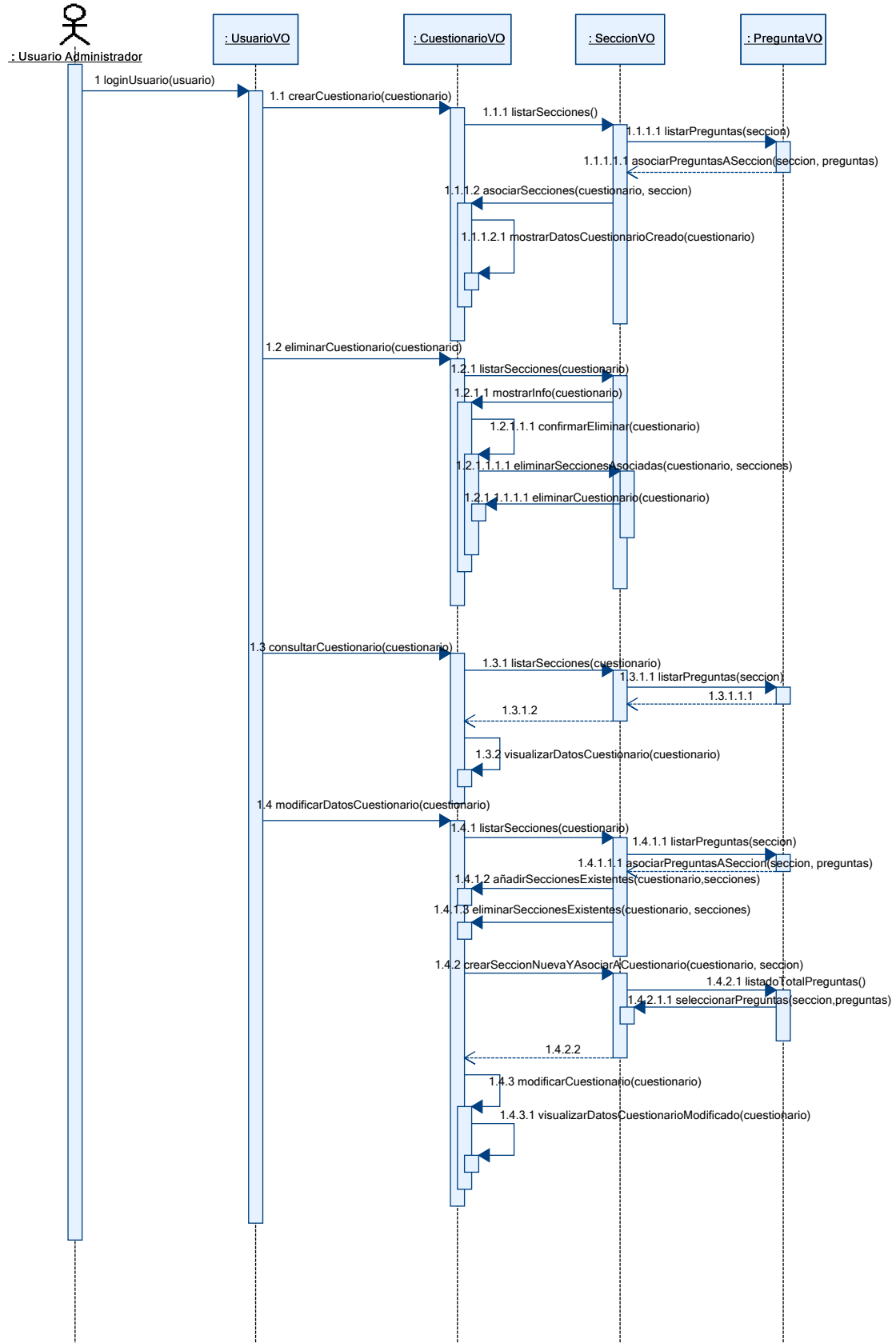


Figura 39. Diagrama de secuencia- Gestión de cuestionarios

• Gestión de secciones

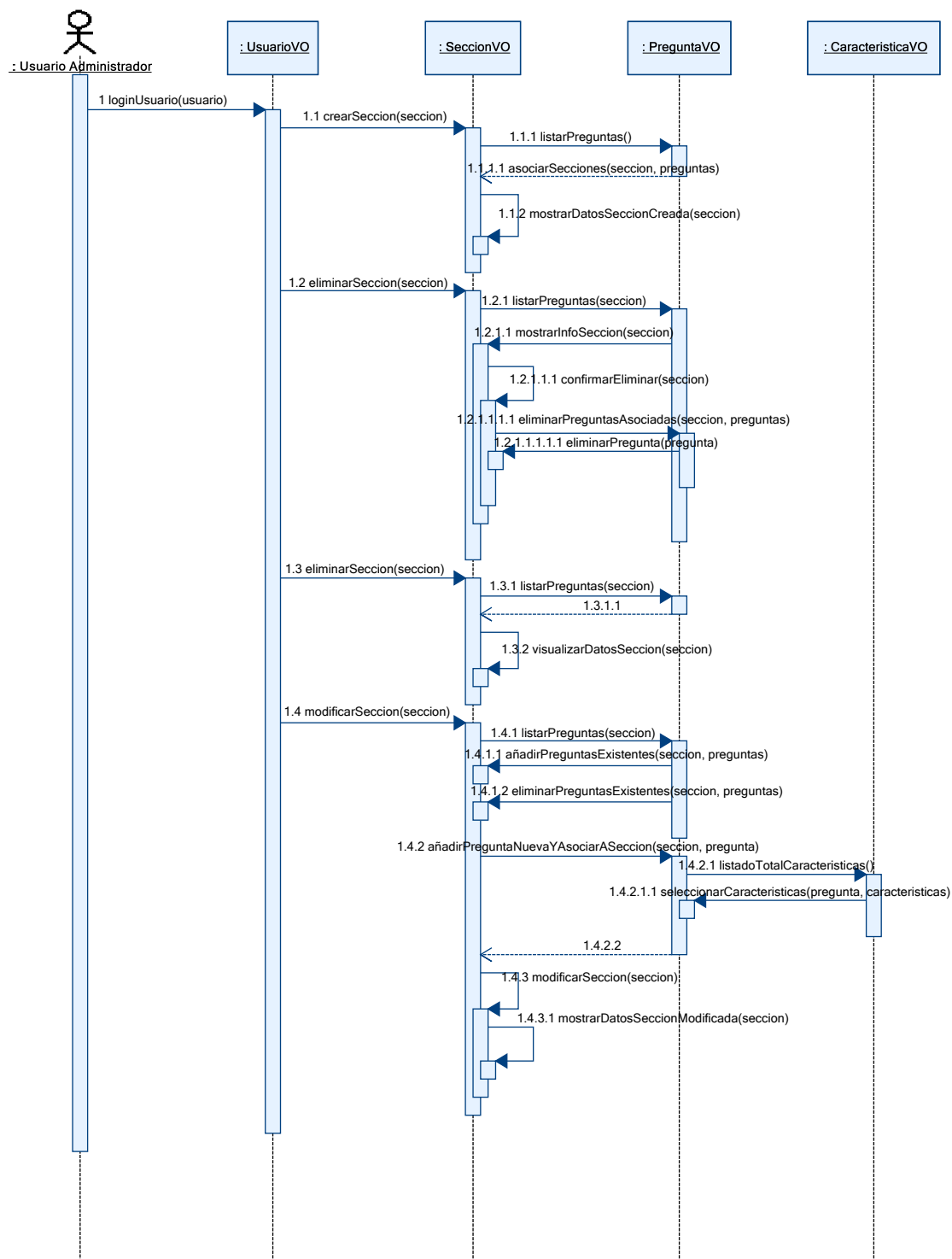


Figura 40. Diagrama de secuencia- Gestión de secciones

• Gestión de preguntas

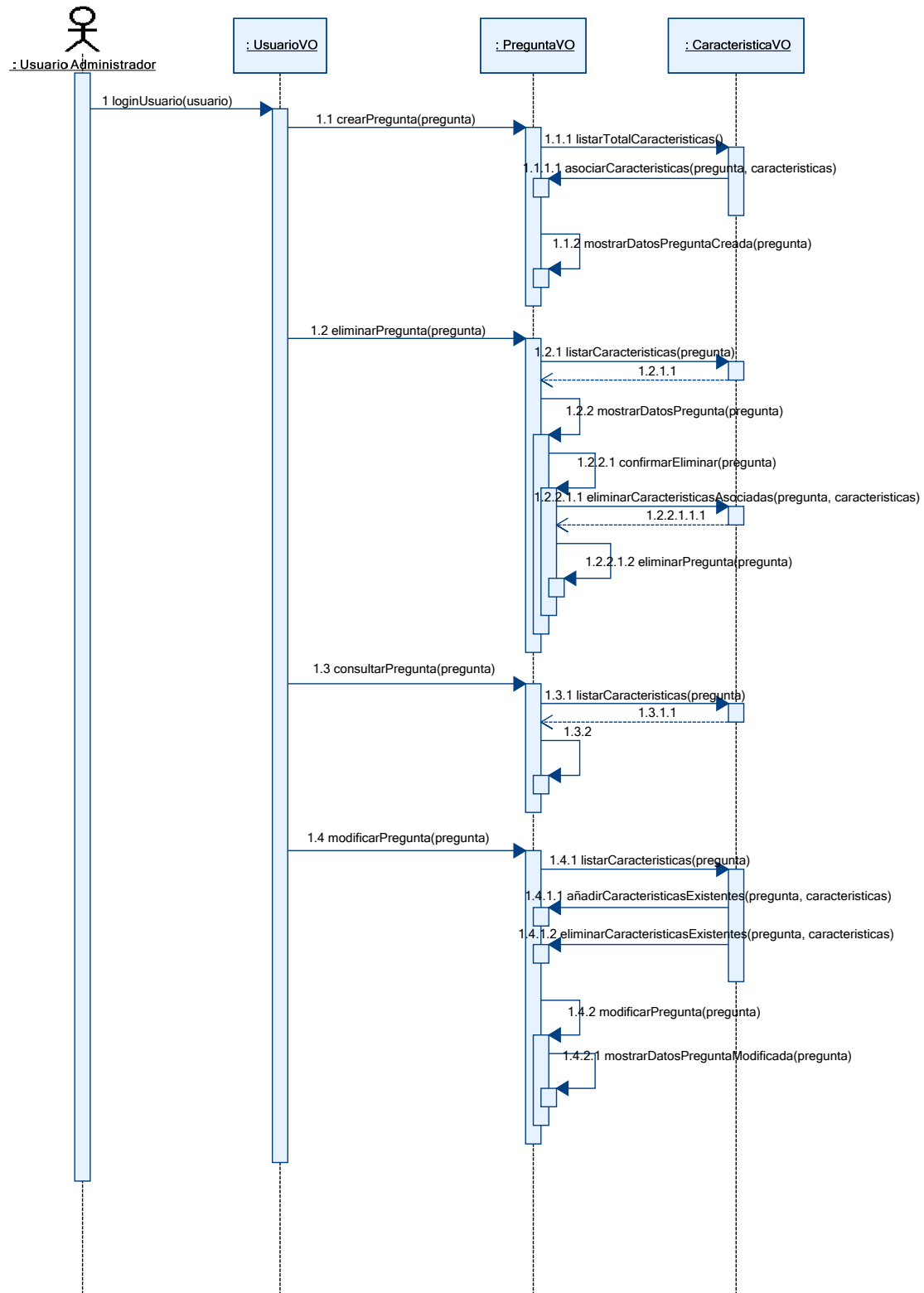


Figura 41. Diagrama de secuencia- Gestión de preguntas

• Visualización de estadísticas

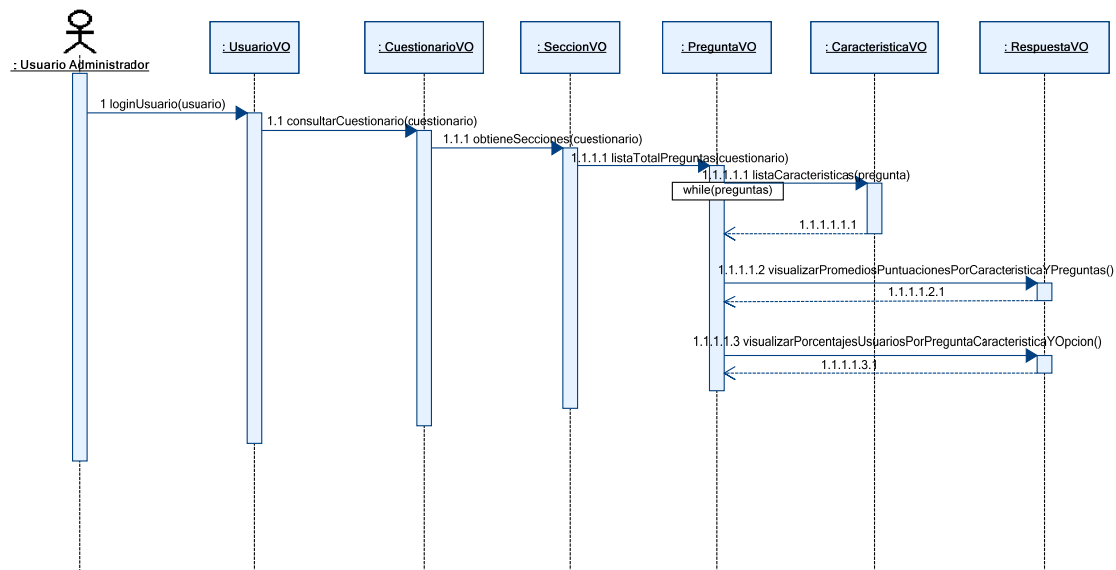


Figura 42. Diagrama de secuencia- Visualización de estadísticas

5.4. Diseño de la interfaz gráfica

La interfaz gráfica de la aplicación está formada por las diversas vistas que componen la arquitectura MVC empleada. Estas vistas son un conjunto de JSPs (*Java Server Pages*). Las JSPs permiten separar la parte dinámica de la página web del código estático. Se tienen, por tanto, un conjunto de vistas dinámicas implementadas con HTML más Javascript más DOM [1].

El diseño de la interfaz perseguía una serie de objetivos tales como que permitiera cierta compatibilidad entre los navegadores, especialmente para *Mozilla Firefox* e *Internet Explorer*. Además se buscaba un diseño sencillo para facilitar el uso de la aplicación al usuario.

Las vistas han quedado divididas en siete partes diferentes, intentando optimizar el espacio de las mismas lo máximo posible:

- ✚ **Cabecera:** Contiene el logotipo de la aplicación, un título descriptivo de la misma y el *login* del usuario conectado en ese momento y un enlace para que pueda cerrar sesión cuando lo desee.
- ✚ **Menú de usuario:** Menú que pasando el puntero del ratón sobre él, muestra las funcionalidades permitidas para cada módulo de la aplicación. Dependiendo de si el usuario tiene un rol básico o de administrador, la aplicación cargará un menú de usuario u otro.

- ✚ **Breadcrumb:** En castellano llamada *miga de pan*. Muestra el camino seguido hasta haber llegado a la página actual. En este caso se ha implementado de manera estática, por lo que no ofrece el “camino literal” seguido, sino que muestra uno básico para cada funcionalidad.
- ✚ **Ayuda:** Mostrada a la derecha del contenido. Pulsando sobre ella se despliega un texto de ayuda específico para cada funcionalidad. Esto aumenta la usabilidad de la aplicación.
- ✚ **Contenido:** Muestra el contenido específico para cada funcionalidad de la aplicación.
- ✚ **Pie de página:** Contiene información relativa a las autoras de la herramienta. Es el mismo para todas las páginas.

El diseño final de la interfaz se muestra en la Figura 43:

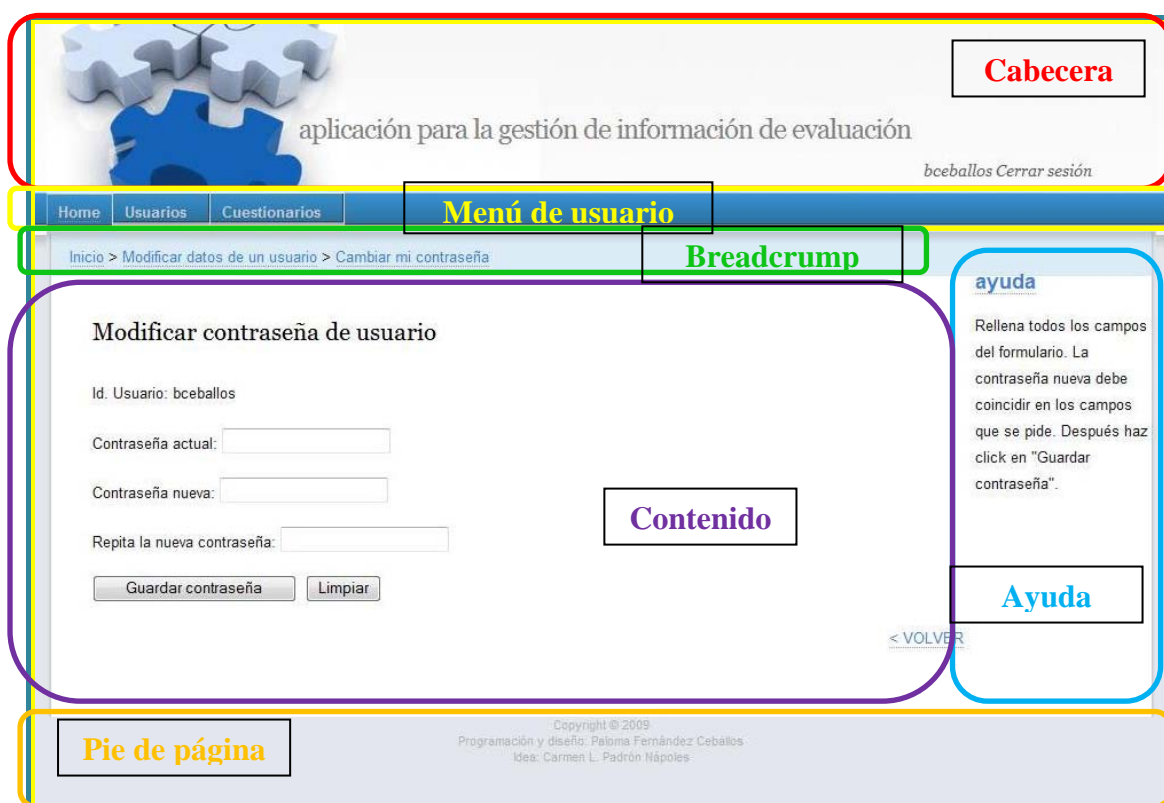


Figura 43. Diseño de la interfaz gráfica



6. MANUAL DE USUARIO

En este capítulo se recoge una guía de instalación para poder ejecutar correctamente la aplicación, así como la secuencia de pasos que deben realizarse para cada una de las funcionalidades que forman parte de la aplicación.

El manual de uso proporcionado contiene los pasos a seguir tanto para arrancar la aplicación como para probar cada una de las funcionalidades de la misma, estas son, completar un cuestionario, visualizar estadísticas, gestionar usuarios, gestionar secciones y preguntas así como el resto de funcionalidades como acceder a la aplicación mediante *login* y *password*.

6.1. Manual de configuración de la aplicación

El presente documento contiene en los siguientes apartados el diagrama de despliegue de la aplicación y los pasos a seguir para la instalación y el correcto funcionamiento de la aplicación.

6.1.1. Diagrama de despliegue

Los componentes del sistema se situarán en distintos puntos del nivel de la arquitectura, dependiendo de la función que realicen.

Siguiendo el estándar J2EE que permite el desarrollo de aplicaciones empresariales de una manera sencilla y eficiente, mostramos el diagrama de despliegue de la aplicación. En éste se plasman las relaciones físicas entre los componentes hardware y los componentes software del sistema, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución.

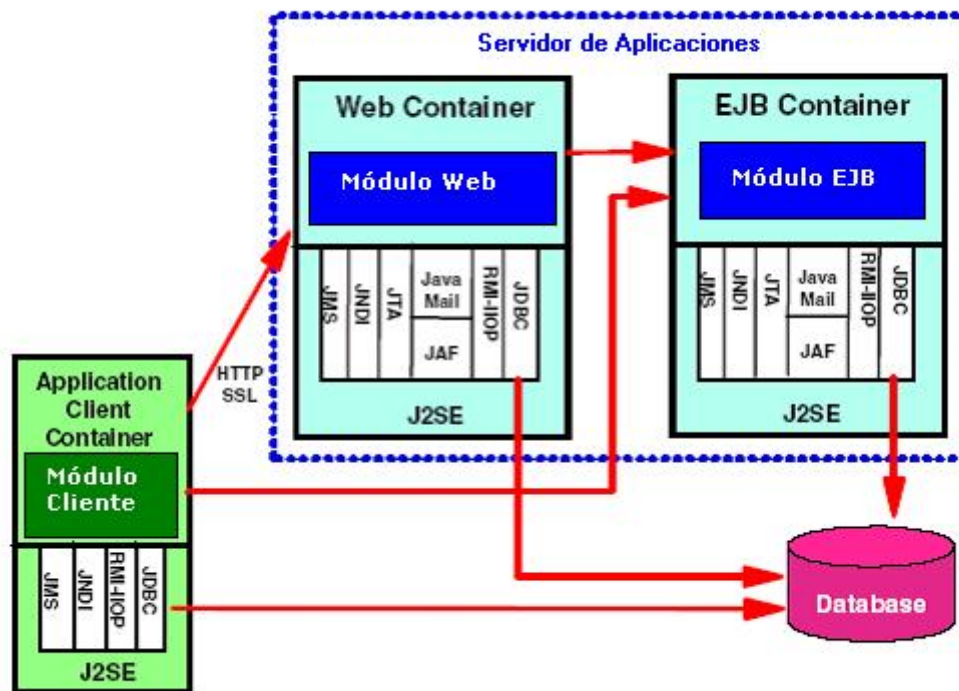


Figura 44. Módulos utilizados arquitectura J2EE

- ✚ **Módulo Web:** Compuesto por: Ficheros *.html, jsp, Servlets, etc
- ✚ **Módulo EJB:** Este módulo está destinado a almacenar EJB propios de la aplicación, estructurado como *.jar con su correspondiente fichero xml de descripción.
- ✚ **Utilidades *.jar:** Utilidades compartidas por los diferentes módulos.
- ✚ **Módulo Cliente:** Clases java ejecutadas en el cliente.
- ✚ **Conectores.**

6.1.2. Pasos para la instalación y configuración de la aplicación

A continuación se listan los pasos a seguir para la instalación y configuración de la aplicación.

(1) Instalar y configurar la base de datos

1. Descargar el software necesario para la instalación de la base de datos MySQL 5.0. Esto se puede hacer desde la dirección:

<http://dev.mysql.com/downloads/mysql/5.0.html#win32>

2. En el apartado "Windows Downloads" seleccionar:

Windows ZIP/Setup.EXE (x86) (viene con instalador incluido)

3. Guardar y descomprimir el .zip
4. Ejecutar el setup.exe y seguir las instrucciones del asistente de instalación.

MUY IMPORTANTE A TENER EN CUENTA:

Puerto por defecto: **3306**

Nombre de usuario: **"root"** (el de por defecto)

¡Se debe dejar sin contraseña!

(2) Insertar script de la Base de Datos y conectarse a ella.

El primer paso que debemos seguir es Iniciar el servidor MySQL y cargar el script mediante el comando "SOURCE". Para ello abrimos un terminal y escribimos la ruta en la que tenemos guardado el script de la base de datos, en este caso: (asegurarse de que la ruta es correcta).

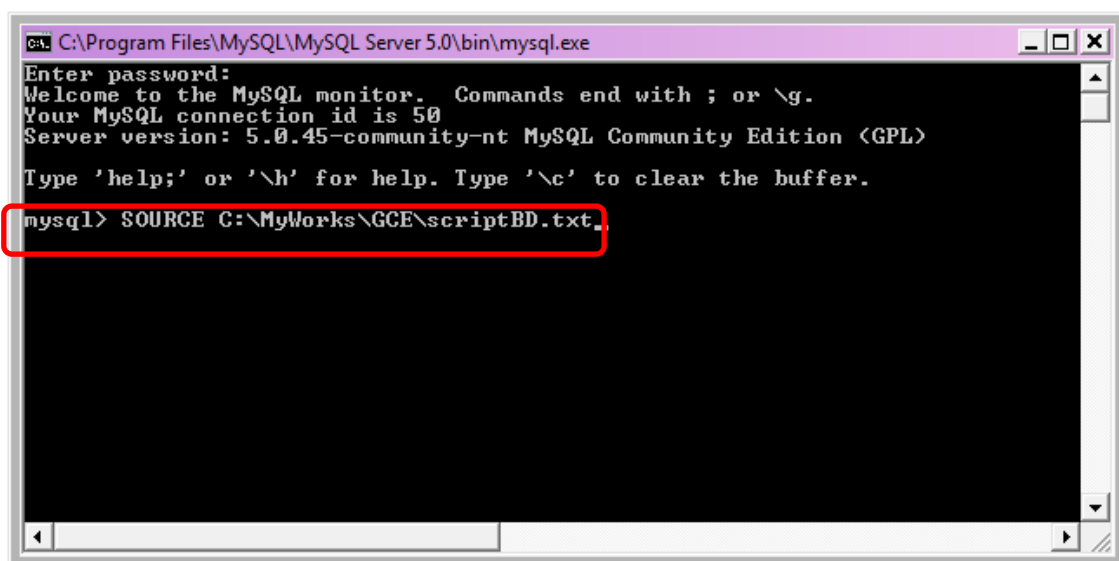


Figura 45. Inserción script base de datos

En lugar de escribir la ruta se puede insertar directamente el script de la base de datos en la consola de MySQL.

El script de la base de datos es el que sigue a continuación:

```
create database BD_GCE;
```

```
use BD_GCE;
```



```
create table AreasInteres (
    id_area varchar(255) primary key,
    area_interes varchar(255)
);

create table CategoriasEspecializacion (
    id_categoria varchar(255) primary key,
    categoria_especializacion varchar(255)
);

create table TiposUsuario(
    id_tipo varchar(255) primary key,
    tipo_usuario varchar(255)
);

create table Usuario (
    login varchar(255) not null,
    password varchar(255) not null,
    nombre varchar(255) not null,
    apellidos varchar(255) not null,
    area_interes varchar(255),
    categoria_especializacion varchar(255),
    tipo varchar(255),
    primary key(login),
    foreign key (tipo) references TiposUsuario (id_tipo)
    on delete restrict on update cascade,
    foreign key (area_interes) references AreasInteres (id_area)
    on delete restrict on update cascade,
    foreign key (categoria_especializacion) references
CategoriasEspecializacion (id_categoria)
    on delete restrict on update cascade
);

create table TipoRespuestas(
    id_tipoRespuestas varchar(255) not null,
    descripcion varchar(255) not null,
    primary key (id_tipoRespuestas)
);

create table ValoresTipoRespuestas(
    id_tipoRespuestas varchar(255) not null,
    puntuacion integer not null,
    descripcion varchar(255) not null,
    primary key (id_tipoRespuestas, puntuacion),
    foreign key (id_tipoRespuestas) references TipoRespuestas
(id_tipoRespuestas)
    on delete cascade on update cascade
);
```



```
create table Cuestionario (  
    id_cuestionario varchar(255) not null,  
    titulo varchar(255) not null,  
    primary key(id_cuestionario)  
);  
  
create table Seccion (  
    id_seccion varchar(255) not null,  
    titulo varchar(255) not null,  
    primary key(id_seccion)  
);  
  
create table Pregunta (  
    id_pregunta varchar(255) not null,  
    titulo varchar(255) not null,  
    tipo_respuestas varchar(255) not null,  
    primary key(id_pregunta),  
    foreign key (tipo_respuestas) references TipoRespuestas  
(id_tipoRespuestas)  
    on delete restrict on update cascade  
);  
  
create table ListaCaracteristicas (  
    id_caracteristica varchar(255),  
    caracteristica varchar(255),  
    primary key(id_caracteristica)  
);  
  
create table SeccionesCuestionario (  
    id_cuestionario varchar(255) not null,  
    id_seccion varchar(255) not null,  
  
    primary key(id_cuestionario, id_seccion),  
    foreign key (id_cuestionario) references Cuestionario  
(id_cuestionario)  
    on delete cascade on update cascade,  
    foreign key (id_seccion) references Seccion (id_seccion)  
    on delete restrict on update cascade  
);  
  
create table PreguntasSeccion (  
    id_seccion varchar(255) not null,  
    id_pregunta varchar(255) not null,  
  
    primary key(id_seccion, id_pregunta),
```



```
foreign key (id_seccion) references Seccion (id_seccion)
on delete cascade on update cascade,
foreign key (id_pregunta) references Pregunta (id_pregunta)
on delete restrict on update cascade
);

create table CaracteristicasPregunta (
    id_pregunta varchar(255) not null,
    id_caracteristica varchar(255) not null,

    primary key(id_pregunta, id_caracteristica),
    foreign key (id_pregunta) references Pregunta (id_pregunta)
on delete cascade on update cascade,
    foreign key (id_caracteristica) references ListaCaracteristicas
(id_caracteristica)
on delete restrict on update cascade
);

create table RespuestasUsuarios (
    id_respuesta varchar(255) primary key,
    login_usuario varchar(255) not null,
    id_cuestionario varchar(255) not null,
    id_seccion varchar(255) not null,
    id_pregunta varchar(255) not null,
    id_caracteristica varchar(255) not null,
    puntuacion_respuesta integer not null,
    comentarios varchar(255),

    foreign key (login_usuario) references Usuario (login)
on delete cascade on update cascade
);
```

Tabla 39. Script de la base de datos

Para conectarse a la base de datos: "connect bd_gce;". El nombre de la base de datos es "bd_gce", éste ha de ser siempre el mismo, no puede variar.

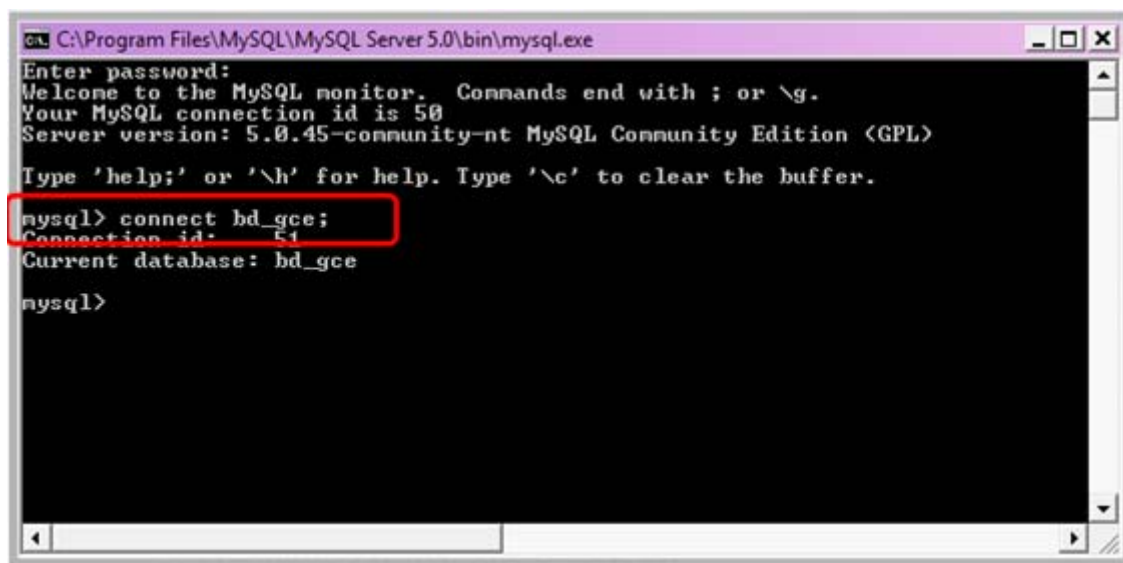


Figura 46. Conexión a la base de datos

(3) Instalar y arrancar JBoss

1. Descargar el software del servidor de aplicaciones JBoss 4.2.1.GA en la siguiente dirección: <http://labs.jboss.com/jbossas/downloads>
2. Guardar y descomprimir el .zip, de nombre **jboss-4.2.1.GA.zip**.
3. Guardar el archivo del proyecto de nombre **GCE.war** en la carpeta "deploy" del servidor:

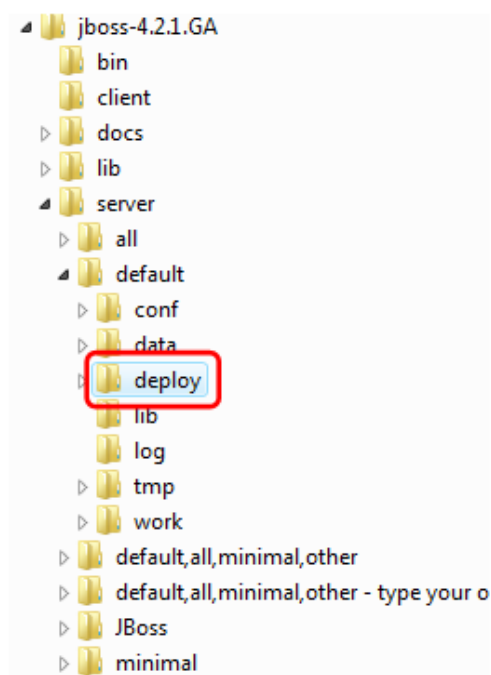


Figura 47. Estructura de carpetas de JBoss

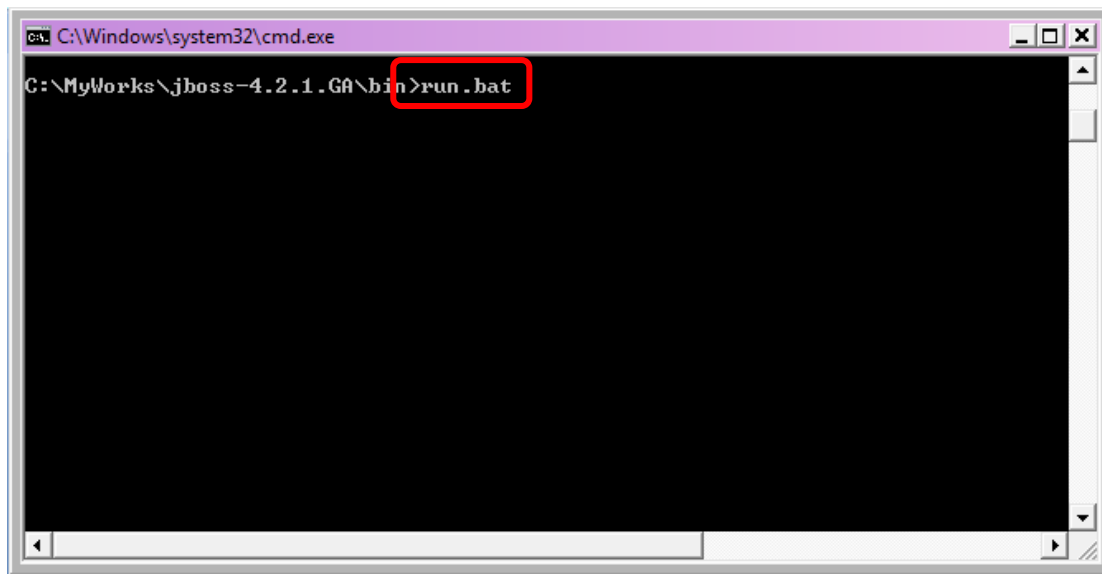


Figura 48. Iniciar JBoss

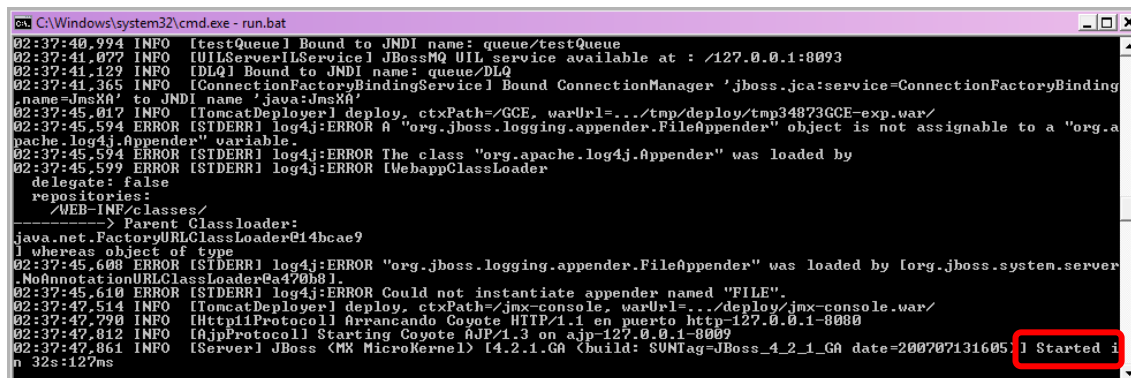


Figura 49. JBoss iniciado

Si hay algún problema al iniciar JBoss [6], se deberán comprobar que las variables de entorno del sistema están configuradas correctamente.

Para Windows Vista habría que ir a Panel de Control > Sistema y mantenimiento > Sistema > Configuración avanzada del sistema > Variables de entorno

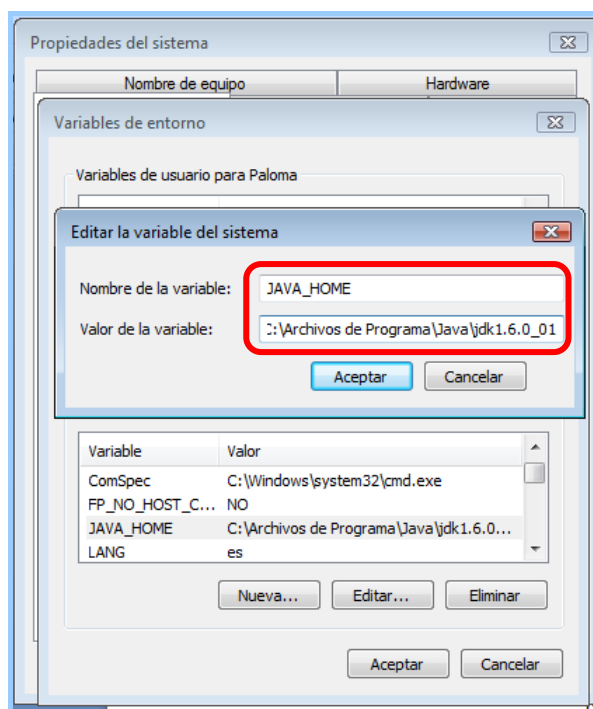


Figura 50. Configuración variables de entorno del sistema

Una vez se tiene instalado el gestor de base de datos, creada la base de datos y arrancado el servidor de aplicaciones, se pasará al siguiente apartado de este capítulo: “Acceso a la aplicación y menú principal”.

6.2. Acceso a la aplicación y menú principal

Para acceder a la aplicación es necesario abrir un navegador de Internet y teclear la siguiente url: <http://md2.dei.inf.uc3m.es:8080/GCE>.

La Figura 53 presenta la página de bienvenida al sistema. Para acceder a la aplicación, el usuario debe tener asignados previamente un login y una contraseña que deberá introducir en este formulario.

Una vez logueado el usuario, se cargará la página principal con un menú desplegable en la parte superior que da acceso a las distintas funcionalidades.

Dependiendo del tipo de usuario que se tenga, “normal” o “administrador”, la aplicación cargará un menú desplegable u otro:

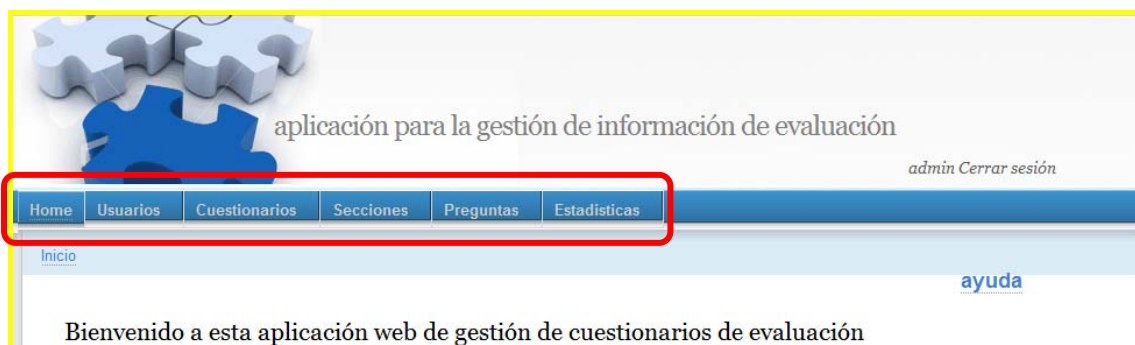


Figura 51. Menú administrador

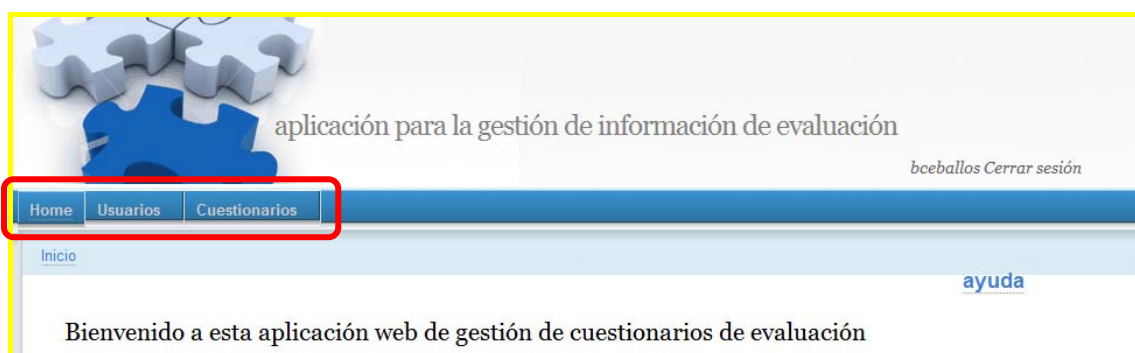


Figura 52. Menú usuario básico

6.3. Tipos de usuarios

Volviendo a la sección anterior, se tienen pues, dos tipos de usuarios: Usuario "normal" o "básico" y usuario "administrador".

El usuario habitual de la aplicación será el "básico", y podrá modificar sus datos personales o bien elegir un cuestionario y responderlo.

El usuario administrador se encargará de la gestión de usuarios y cuestionarios, y además podrá visualizar estadísticas basadas en las respuestas de los usuarios.

La administración de dichos usuarios, la lleva por tanto el administrador, el cual podrá dar de alta, de baja, modificar los datos de un usuario específico o consultarlos simplemente.

6.4. Operaciones básicas

Las operaciones descritas a continuación proporcionan la funcionalidad básica de la aplicación:

6.4.1. Loguearse

Para entrar en la aplicación, se deberá introducir el *login* de usuario y la contraseña de acceso, después pulsar el botón de “Enviar”.

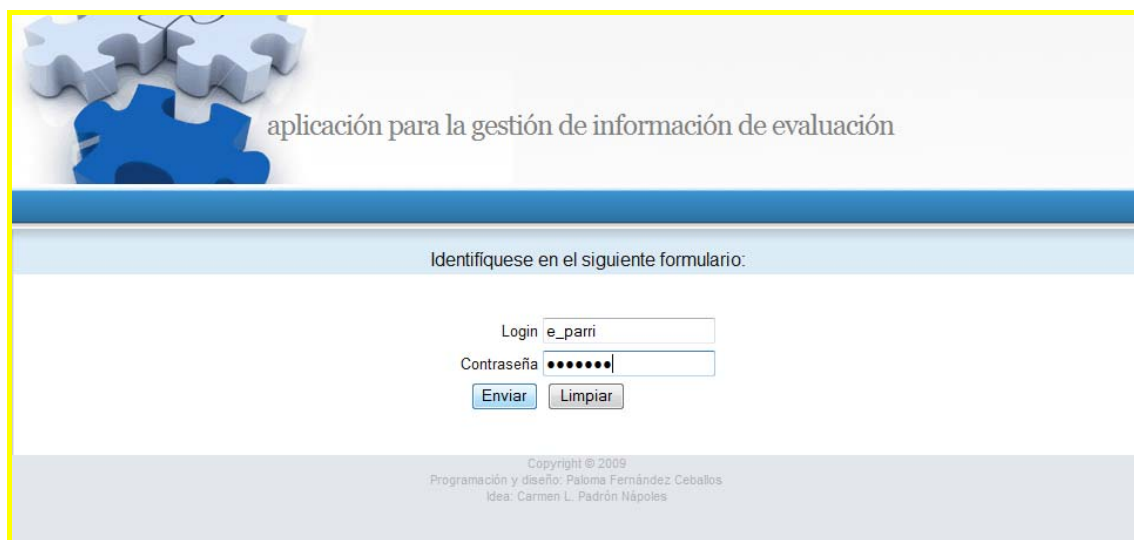


Figura 53. Página para loguearse

6.4.2. Cerrar sesión

Para salir de la aplicación correctamente, deberá hacer clic sobre el enlace “Cerrar sesión”.

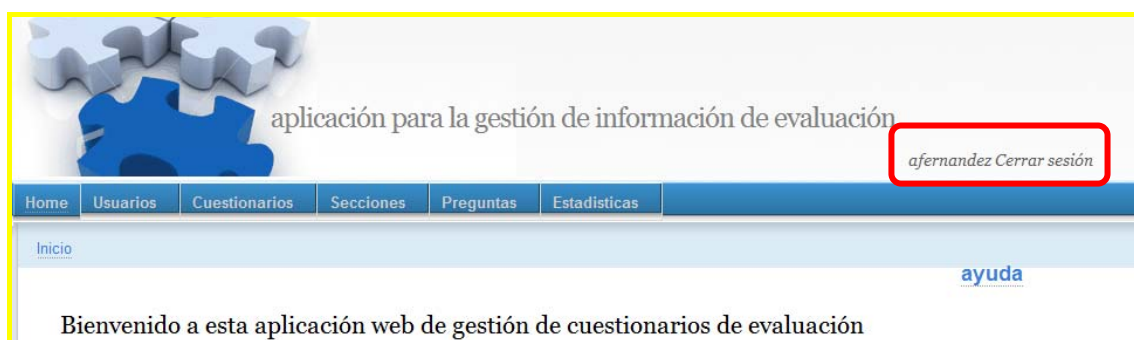


Figura 54. Página para cerrar sesión

6.4.3. Modificar datos de usuario

Para modificar los datos de un usuario, se debe seguir la siguiente secuencia de pasos:

- 1) Seleccionar “Modificar datos de usuario” en el menú desplegable de arriba:

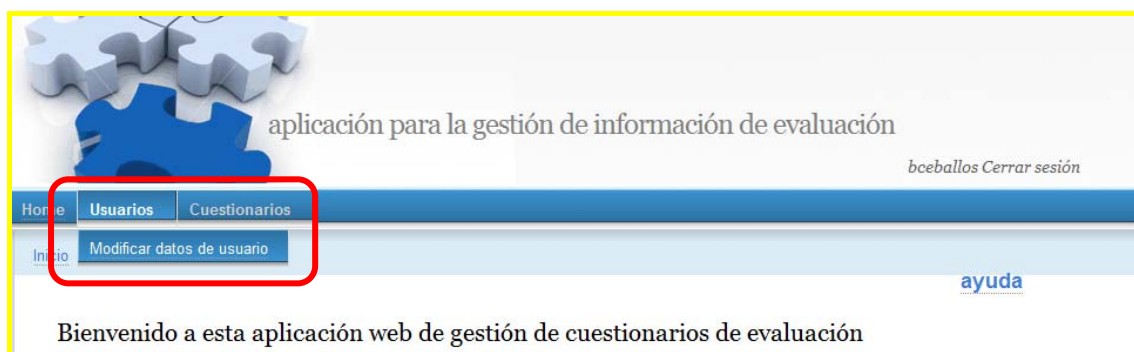


Figura 55. Página que muestra como elegir una opción del menú

- 2) Seleccionar el *login* de usuario de la lista desplegable y pulsar “Mostrar datos de usuario”:



Figura 56. Página para modificar los datos de un usuario

Si se quiere modificar la contraseña de usuario, hacer clic sobre “Modificar contraseña”, se introduce la contraseña actual y la contraseña

nueva dos veces y “Guardar contraseña”. En caso de ser tener rol de administrador, no se pedirá la contraseña actual.

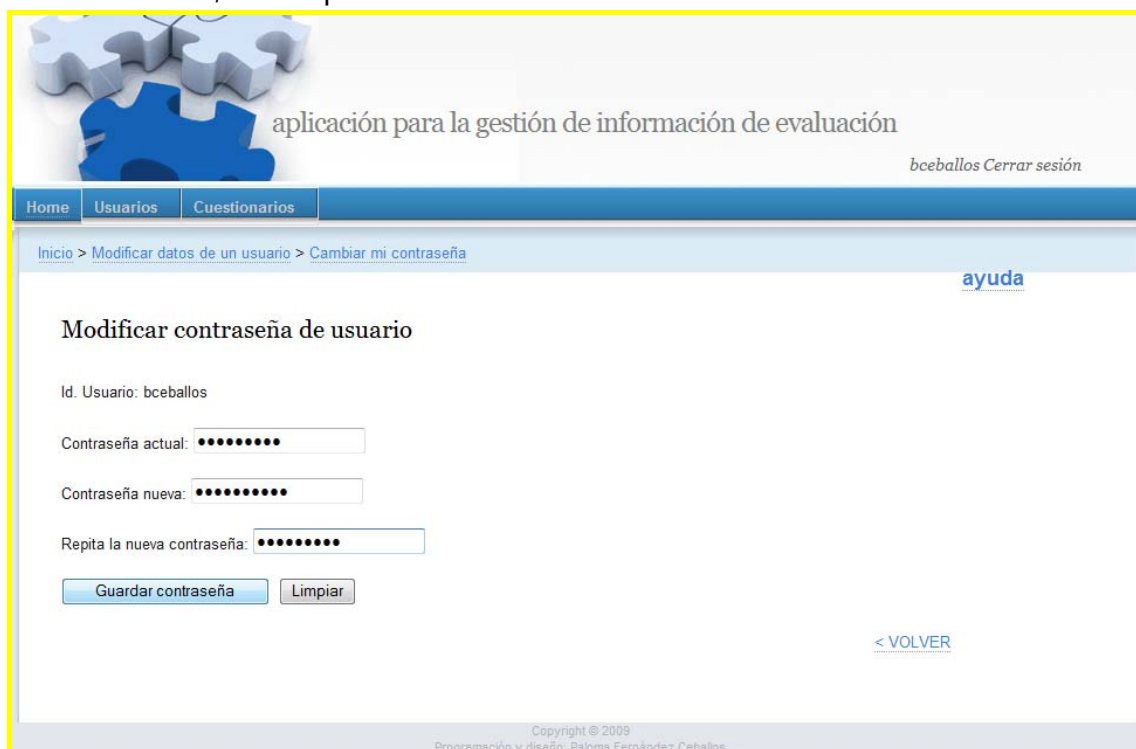


Figura 57. Página para modificar la contraseña de usuario

6.4.4. Responder un cuestionario

Al igual que antes, se despliega la opción de “Cuestionarios” en el menú de usuario y se selecciona “Responder cuestionario”:

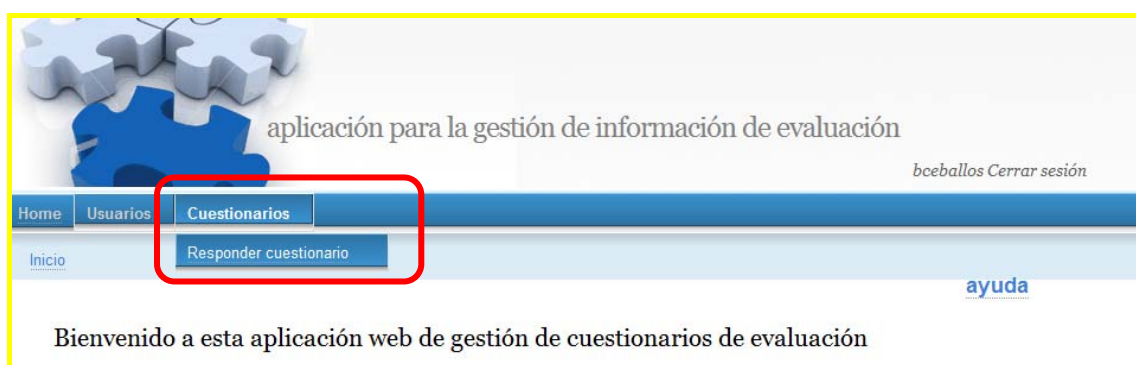
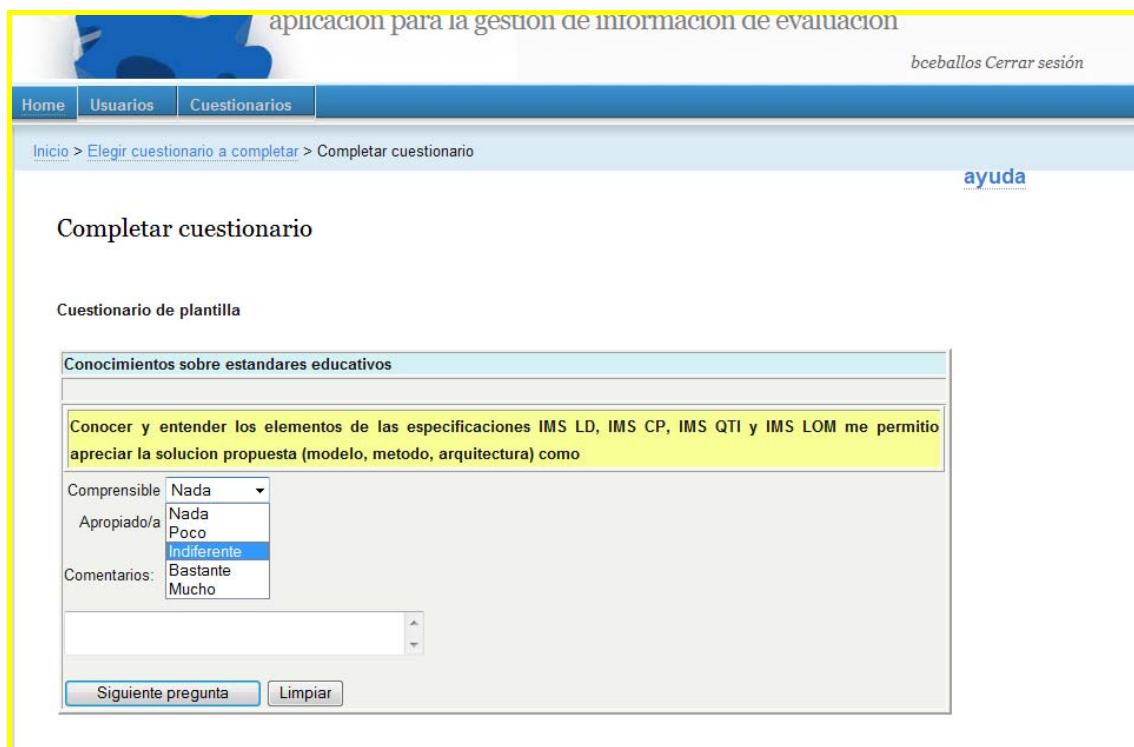


Figura 58. Página para elegir la opción de responder un cuestionario

En el desplegable que aparece, se elige el cuestionario filtrando por su título, se hace *click* sobre la opción de mostrar datos del cuestionario y acto seguido se va a “Responder cuestionario”.

Aparecerá un formulario, con los datos del cuestionario, la primera pregunta a responder, las características asociadas a la misma cada una de ellas con un desplegable con una serie de puntuaciones permitidas.



aplicación para la gestión de información de evaluación

bceballos Cerrar sesión

Home Usuarios Cuestionarios

Inicio > Elegir cuestionario a completar > Completar cuestionario

ayuda

Completar cuestionario

Cuestionario de plantilla

Conocimientos sobre estándares educativos

Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitió apreciar la solución propuesta (modelo, método, arquitectura) como

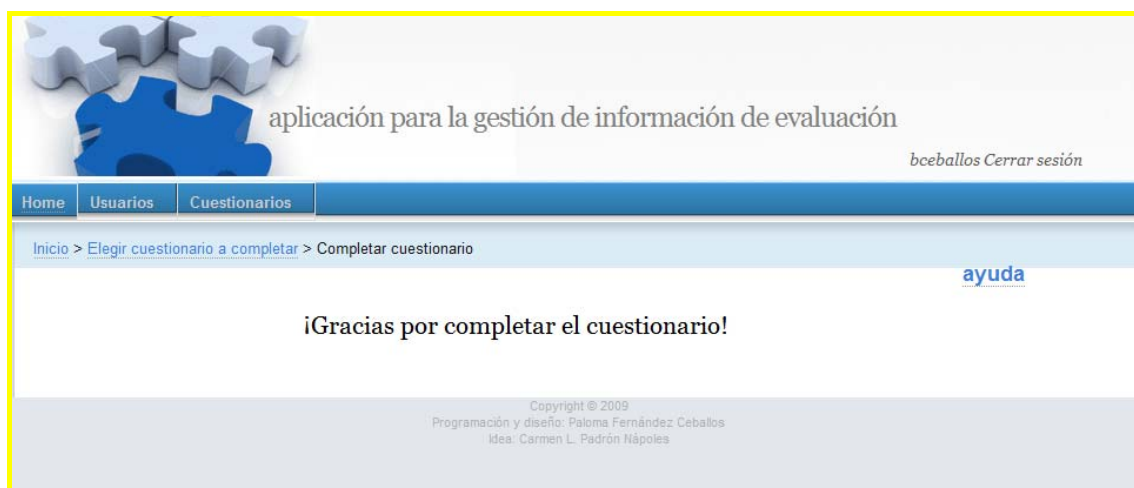
Comprensible: Nada, Apropiado/a: Nada, Poco, Indiferente, Bastante, Mucho

Comentarios:

Siguiente pregunta Limpiar

Figura 59. Página para responder una pregunta del cuestionario

Para completar el cuestionario correctamente, se empezará por la primera pregunta en adelante, eligiendo las puntuaciones convenientes por pregunta y característica y pulsando el botón "Contestar siguiente" hasta llegar al fin del cuestionario, que vendrá definido de esta forma:



aplicación para la gestión de información de evaluación

bceballos Cerrar sesión

Home Usuarios Cuestionarios

Inicio > Elegir cuestionario a completar > Completar cuestionario

ayuda

¡Gracias por completar el cuestionario!

Copyright © 2009
Programación y diseño: Paloma Fernández Ceballos
Idea: Carmen L. Padrón Nápoles

Figura 60. Página que muestra que el cuestionario respondido con éxito



6.5. Operaciones exclusivas para el administrador

Un usuario administrador además de las operaciones definidas en la sección anterior tendrá acceso a las funcionalidades descritas a continuación, las cuales están divididas en cinco grupos:

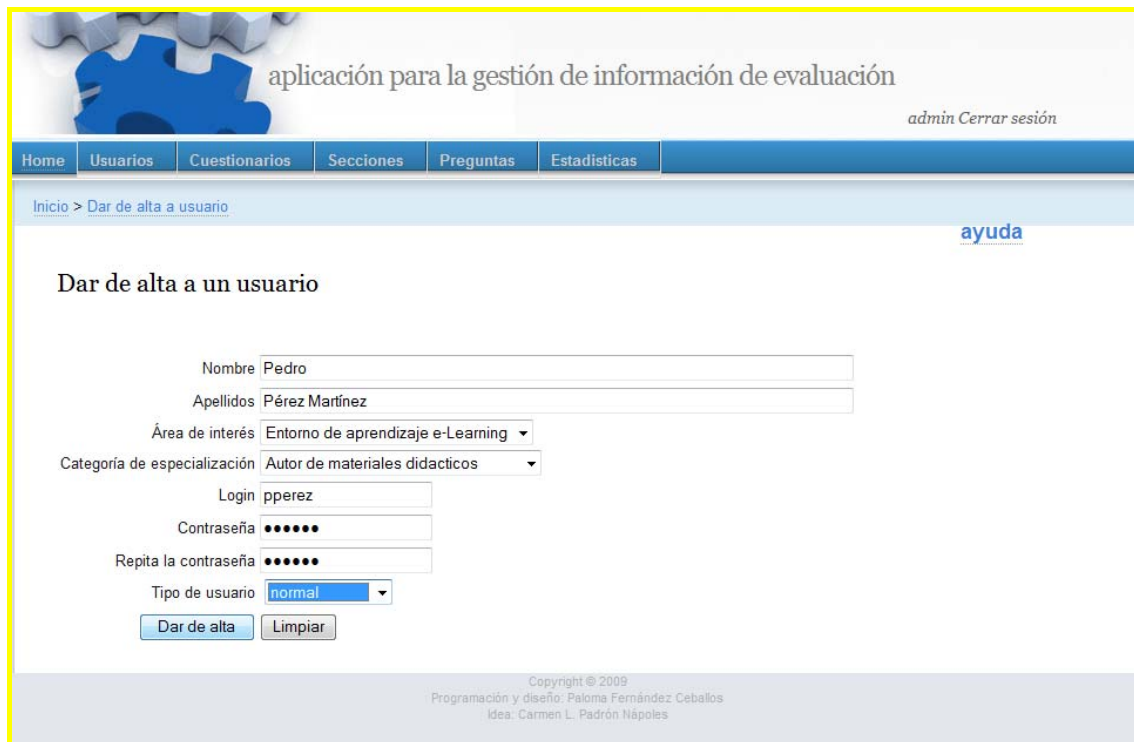
- ✚ **Gestión de usuarios:** Engloba las operaciones de dar de alta a un usuario, dar de baja, y consultar o modificar los datos de un usuario.
- ✚ **Gestión de cuestionarios:** Permite crear un cuestionario, modificarlo, eliminarlo, consultarlo y responderlo.
- ✚ **Gestión de secciones:** Permite crear, modificar, consultar o eliminar secciones.
- ✚ **Gestión de preguntas:** Engloba la creación, modificación, consulta y eliminación de preguntas.
- ✚ **Estadísticas:** Permite el acceso a informes estadísticos basados en las respuestas de los usuarios.

6.5.1. Gestión de usuarios

Dar de alta a un usuario:

- 1) Dentro del apartado "Usuarios" elegir la opción "Dar de alta a usuario" y se cargará una interfaz como la de la Figura 61.
- 2) Se deben rellenar los campos con los datos pertinentes y a continuación pulsar sobre el botón de "Dar de alta".
- 3) Para finalizar, la aplicación muestra un mensaje de éxito de la operación que contiene los datos del usuario creado.

Las Figuras 61 y 62 muestran la secuencia de pasos a seguir:



aplicación para la gestión de información de evaluación

admin Cerrar sesión

Home Usuarios Cuestionarios Secciones Preguntas Estadísticas

Inicio > Dar de alta a usuario ayuda

Dar de alta a un usuario

Nombre

Apellidos

Área de interés

Categoría de especialización

Login

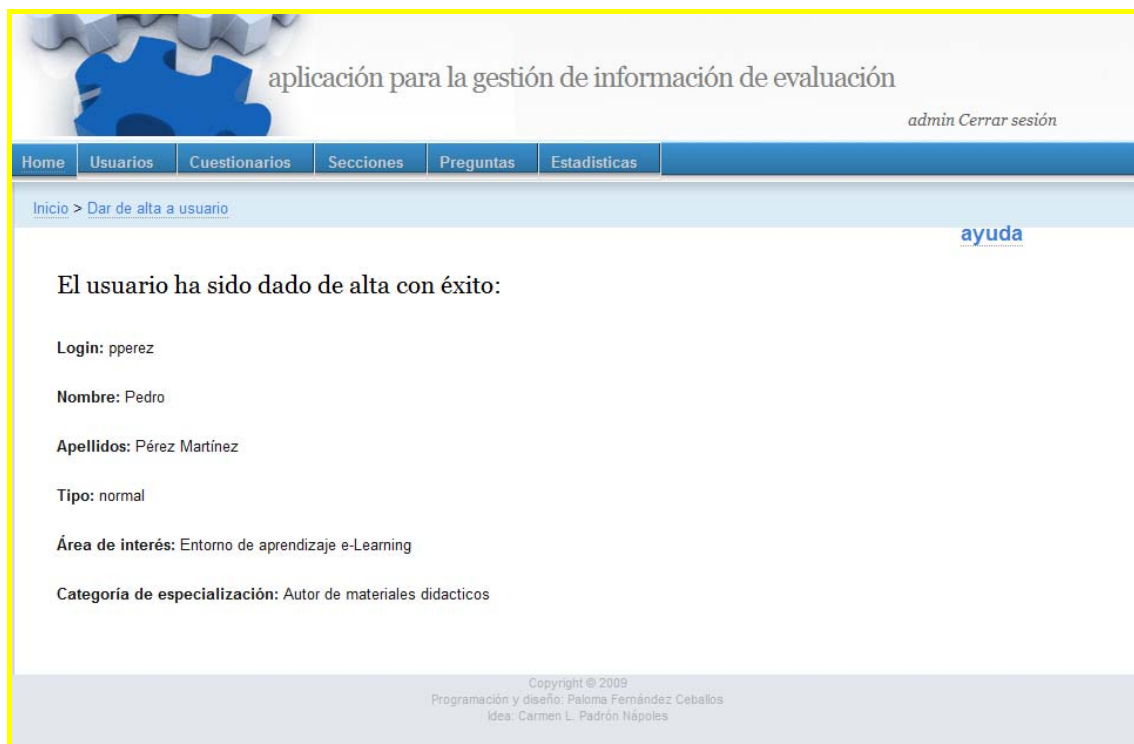
Contraseña

Repita la contraseña

Tipo de usuario

Copyright © 2009
Programación y diseño: Paloma Fernández Ceballos
Idea: Carmen L. Padrón Nápoles

Figura 61. Página para dar de alta a usuario



aplicación para la gestión de información de evaluación

admin Cerrar sesión

Home Usuarios Cuestionarios Secciones Preguntas Estadísticas

Inicio > Dar de alta a usuario ayuda

El usuario ha sido dado de alta con éxito:

Login: pperez

Nombre: Pedro

Apellidos: Pérez Martínez

Tipo: normal

Área de interés: Entorno de aprendizaje e-Learning

Categoría de especialización: Autor de materiales didacticos

Copyright © 2009
Programación y diseño: Paloma Fernández Ceballos
Idea: Carmen L. Padrón Nápoles

Figura 62. Página que muestra que un usuario ha sido dado de alta con éxito

Consultar los datos de un usuario:

- 1) Elegir la opción "Consultar datos de usuario" del apartado "Usuarios" del menú desplegable.
- 2) Elegir el usuario por su *login* y hacer *click* en "Mostrar información de usuario".

A continuación la aplicación mostrará los datos personales del usuario seleccionado.

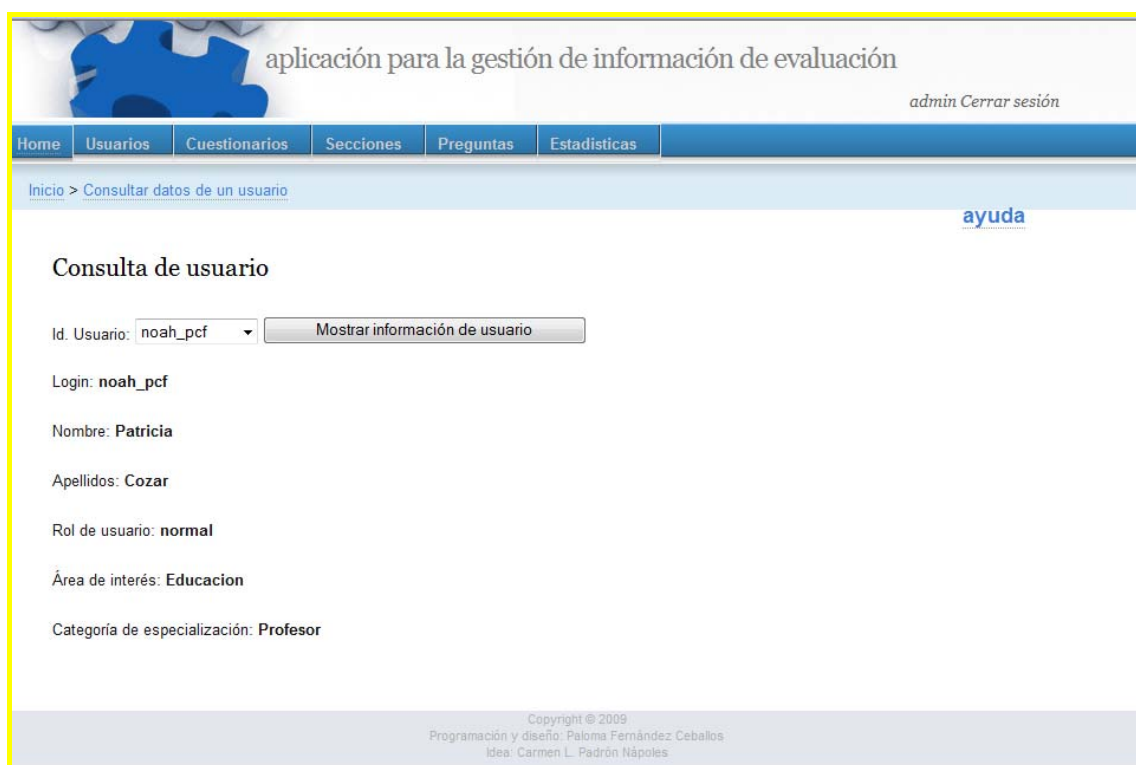


Figura 63. Página para consultar datos de un usuario

Modificar los datos de un usuario:

Ver sección 6.4.3. *Modificar datos de usuario*, apartado de este capítulo donde se explican las funcionalidades permitidas para un usuario básico y que a su vez pueden realizar también los usuarios con rol de administrador.

Dar de baja a un usuario:

- 1) Elegir la opción "Dar de baja a usuario" del apartado "Usuarios" del menú desplegable.
- 2) Elegir el usuario por su *login* y *click* en "Buscar".

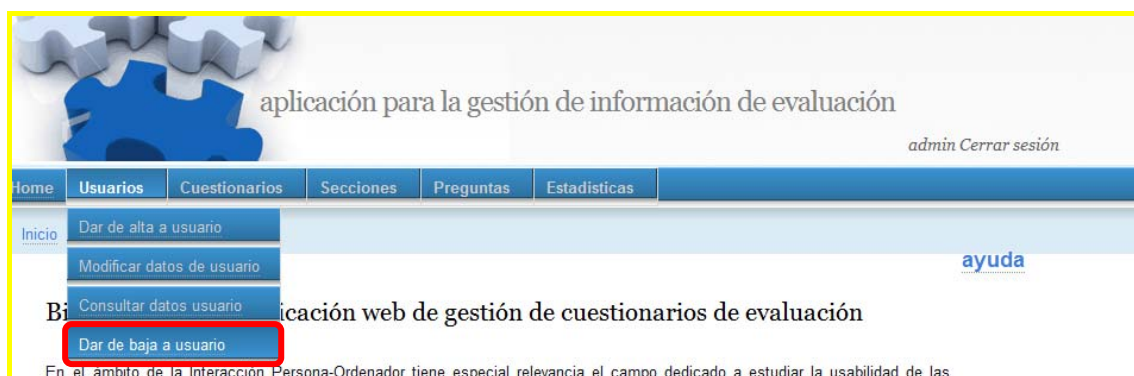


Figura 64. Página para elegir la opción de dar de baja a un usuario



Figura 65. Página para dar de baja a un usuario

3) Hacer *click* en "Dar de baja" y si el usuario se ha dado de baja correctamente de la base de datos, la aplicación mostrará un mensaje de éxito.

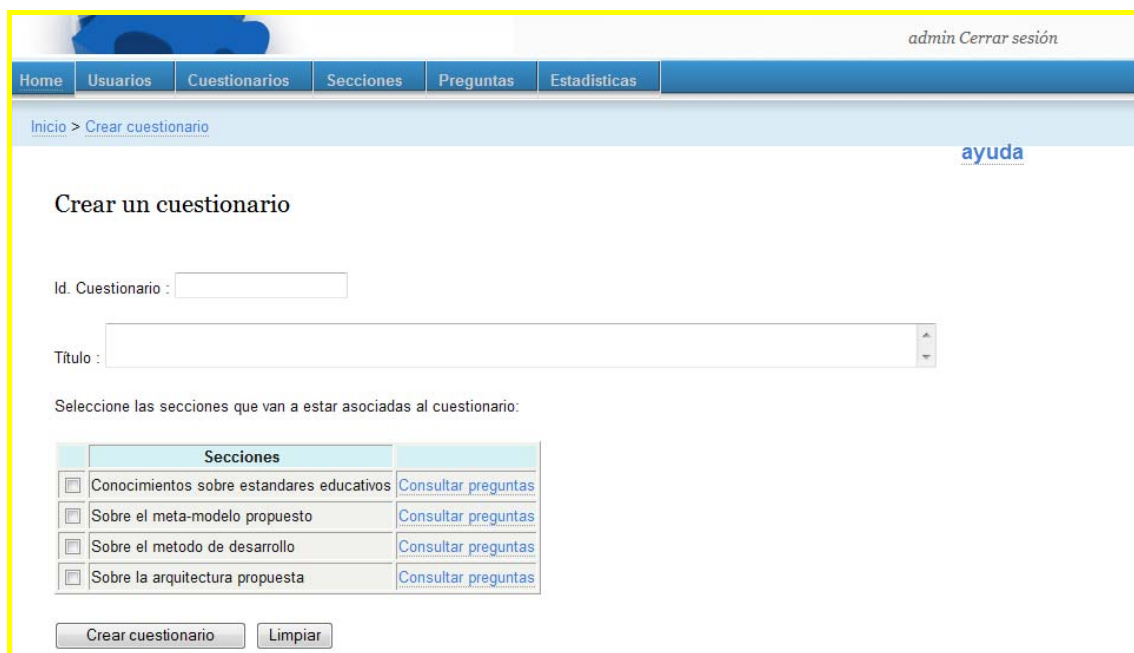
6.5.2. Gestión de cuestionarios

Crear un cuestionario:

1) Dentro del apartado "Cuestionarios" elegir la opción "Crear cuestionario" y se cargará una interfaz como la de la Figura 66.

2) Se deben rellenar los campos con los datos pertinentes y a continuación pulsar en el botón de "Crear cuestionario".

3) Para finalizar, la aplicación muestra un mensaje de éxito de la operación que contiene los datos del cuestionario creado.



Secciones	
<input type="checkbox"/> Conocimientos sobre estandares educativos	Consultar preguntas
<input type="checkbox"/> Sobre el meta-modelo propuesto	Consultar preguntas
<input type="checkbox"/> Sobre el metodo de desarrollo	Consultar preguntas
<input type="checkbox"/> Sobre la arquitectura propuesta	Consultar preguntas

Figura 66. Página para crear un cuestionario

Consultar los datos de un cuestionario:

- 1) Elegir la opción "Consultar cuestionario" del apartado "Cuestionarios" del menú desplegable.
- 2) Elegir el cuestionario por su título y *click* en "Mostrar información de cuestionario".

Las Figuras 67, 68 y 69 muestran un ejemplo de consulta de información de un cuestionario.



aplicación para la gestión de información de evaluación

admin Cerrar sesión

Home Usuarios Cuestionarios Secciones Preguntas Estadísticas

Inicio > Consultar datos cuestionario

ayuda

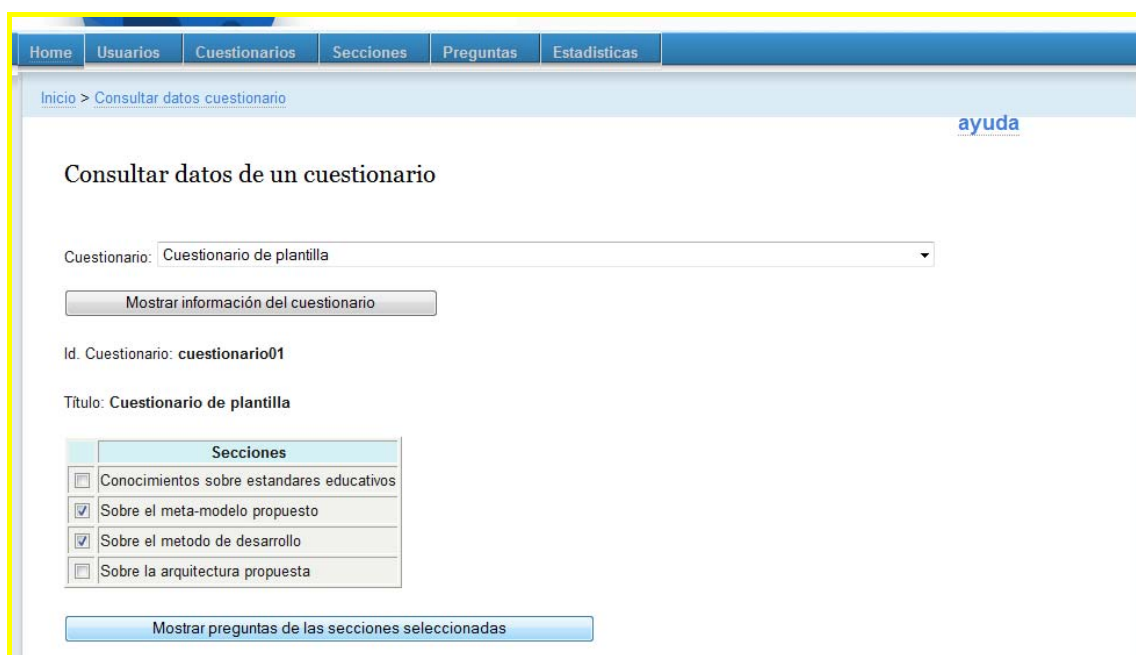
Consultar datos de un cuestionario

Cuestionario: Cuestionario de plantilla

Mostrar información del cuestionario

Copyright © 2009
Programación y diseño: Paloma Fernández Ceballos
Ideas: Carmen L. Padrón Nápoles

Figura 67. Página para consultar un cuestionario



Home Usuarios Cuestionarios Secciones Preguntas Estadísticas

Inicio > Consultar datos cuestionario

ayuda

Consultar datos de un cuestionario

Cuestionario: Cuestionario de plantilla

Mostrar información del cuestionario

Id. Cuestionario: cuestionario01

Título: Cuestionario de plantilla

Secciones	
<input type="checkbox"/>	Conocimientos sobre estándares educativos
<input checked="" type="checkbox"/>	Sobre el meta-modelo propuesto
<input checked="" type="checkbox"/>	Sobre el metodo de desarrollo
<input type="checkbox"/>	Sobre la arquitectura propuesta

Mostrar preguntas de las secciones seleccionadas

Figura 68. Página que muestra los datos del cuestionario consultado

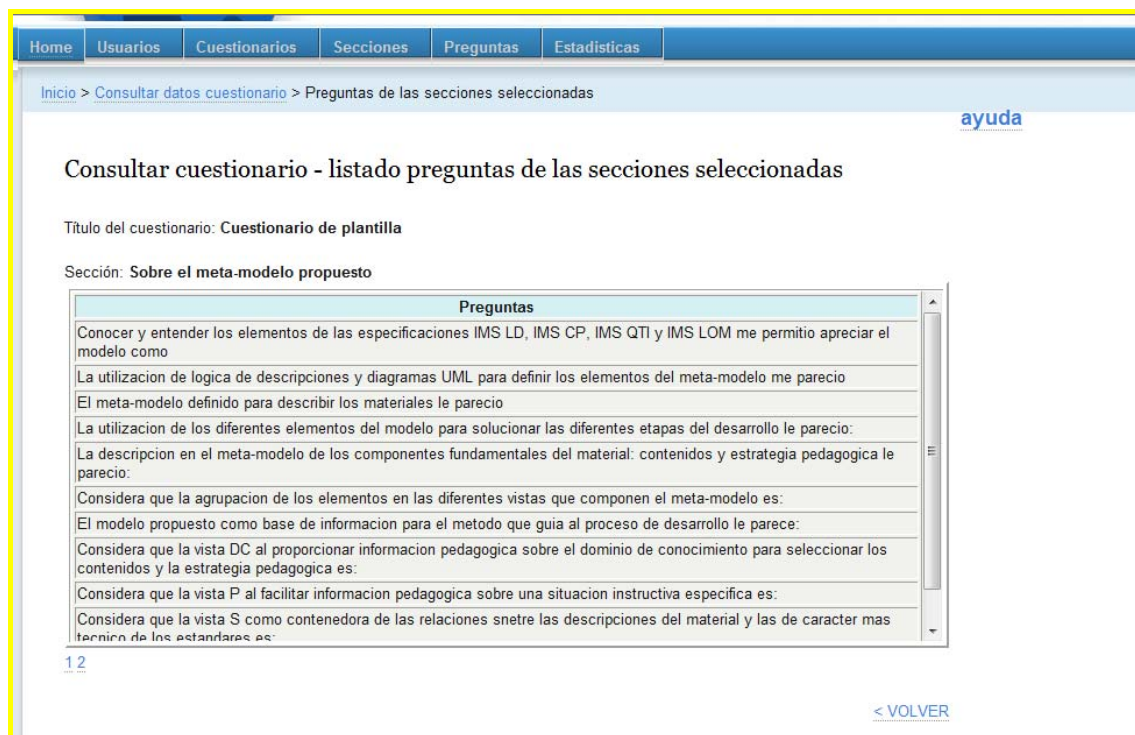


Figura 69. Página que muestra las preguntas de una sección perteneciente a un cuestionario

Modificar los datos de un cuestionario:

- 1) Elegir la opción "Modificar cuestionario" del apartado "Cuestionarios" del menú desplegable.
- 2) Elegir el cuestionario por su título y *click* en "Mostrar información de cuestionario".

Esta funcionalidad permite modificar el título del cuestionario, y las secciones que contiene. Se pueden añadir o eliminar secciones. En caso de añadir secciones, permite tanto añadir secciones previamente existentes en la base de datos como así como crear una sección nueva que se asociará directamente al cuestionario seleccionado.

Las Figuras 70, 71, 72 y 73 muestran un ejemplo de secuencia de pasos a seguir para modificar un cuestionario:



Inicio > [Modificar cuestionario](#)

[ayuda](#)

Modificar datos de un cuestionario

Cuestionario:

Id. Cuestionario: cuestionario01

Título:

Secciones	
Conocimientos sobre estandares educativos	Consultar preguntas
Sobre el meta-modelo propuesto	Consultar preguntas
Sobre el metodo de desarrollo	Consultar preguntas
Sobre la arquitectura propuesta	Consultar preguntas

Figura 70. Página para modificar un cuestionario

Cerrar sesión

Home [Usuarios](#) [Cuestionarios](#) [Secciones](#) [Preguntas](#) [Estadísticas](#)

Inicio > [Modificar cuestionario](#) > [Añadir secciones](#)

[ayuda](#)

Modificar Cuestionario - Añadir secciones

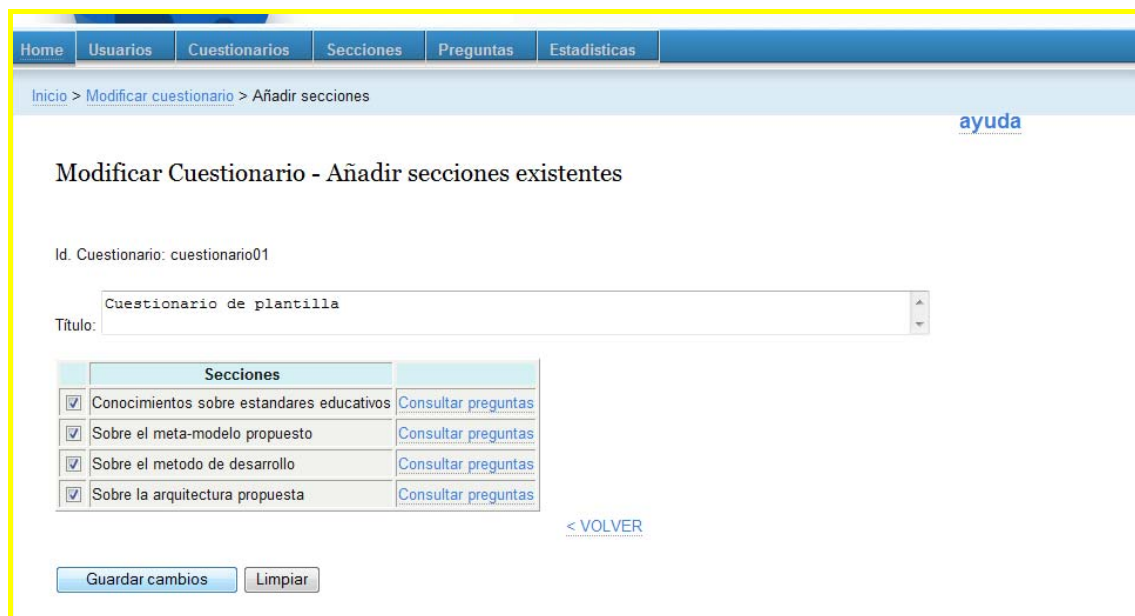
Id. Cuestionario: cuestionario01

Título:

Secciones	
Conocimientos sobre estandares educativos	Consultar preguntas
Sobre el meta-modelo propuesto	Consultar preguntas
Sobre el metodo de desarrollo	Consultar preguntas
Sobre la arquitectura propuesta	Consultar preguntas

[< VOLVER](#)

Figura 71. Página para añadir secciones a un cuestionario



Home Usuarios Cuestionarios Secciones Preguntas Estadísticas

Inicio > Modificar cuestionario > Añadir secciones [ayuda](#)

Modificar Cuestionario - Añadir secciones existentes

Id. Cuestionario: cuestionario01

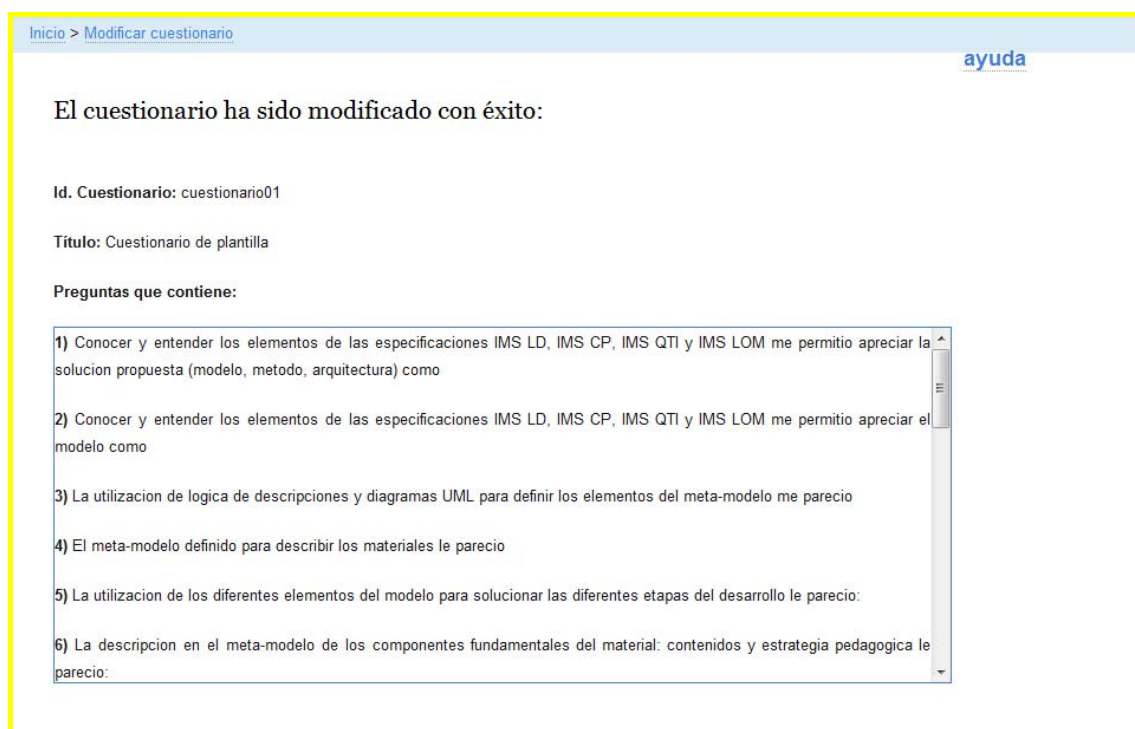
Título: Cuestionario de plantilla

Secciones	
<input checked="" type="checkbox"/> Conocimientos sobre estandares educativos	Consultar preguntas
<input checked="" type="checkbox"/> Sobre el meta-modelo propuesto	Consultar preguntas
<input checked="" type="checkbox"/> Sobre el metodo de desarrollo	Consultar preguntas
<input checked="" type="checkbox"/> Sobre la arquitectura propuesta	Consultar preguntas

< VOLVER

Guardar cambios Limpiar

Figura 72. Página para añadir secciones (previamente creadas) a un cuestionario



Inicio > Modificar cuestionario [ayuda](#)

El cuestionario ha sido modificado con éxito:

Id. Cuestionario: cuestionario01

Título: Cuestionario de plantilla

Preguntas que contiene:

- 1) Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar la solucion propuesta (modelo, metodo, arquitectura) como
- 2) Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar el modelo como
- 3) La utilizacion de logica de descripciones y diagramas UML para definir los elementos del meta-modelo me parecio
- 4) El meta-modelo definido para describir los materiales le parecio
- 5) La utilizacion de los diferentes elementos del modelo para solucionar las diferentes etapas del desarrollo le parecio:
- 6) La descripcion en el meta-modelo de los componentes fundamentales del material: contenidos y estrategia pedagogica le parecio:

< VOLVER

Figura 73. Página que muestra que un cuestionario ha sido modificado con éxito

Eliminar un cuestionario:

- 1) Elegir la opción "Eliminar cuestionario" del apartado "Cuestionarios" del menú desplegable.
- 2) Elegir el cuestionario por su título y *click* en "Mostrar información del cuestionario".

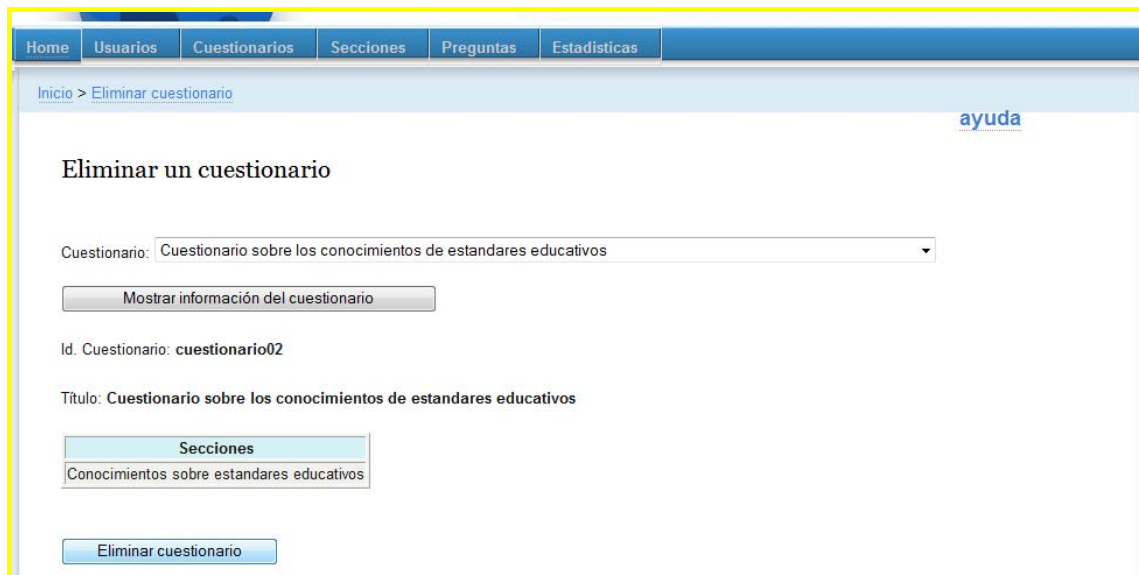


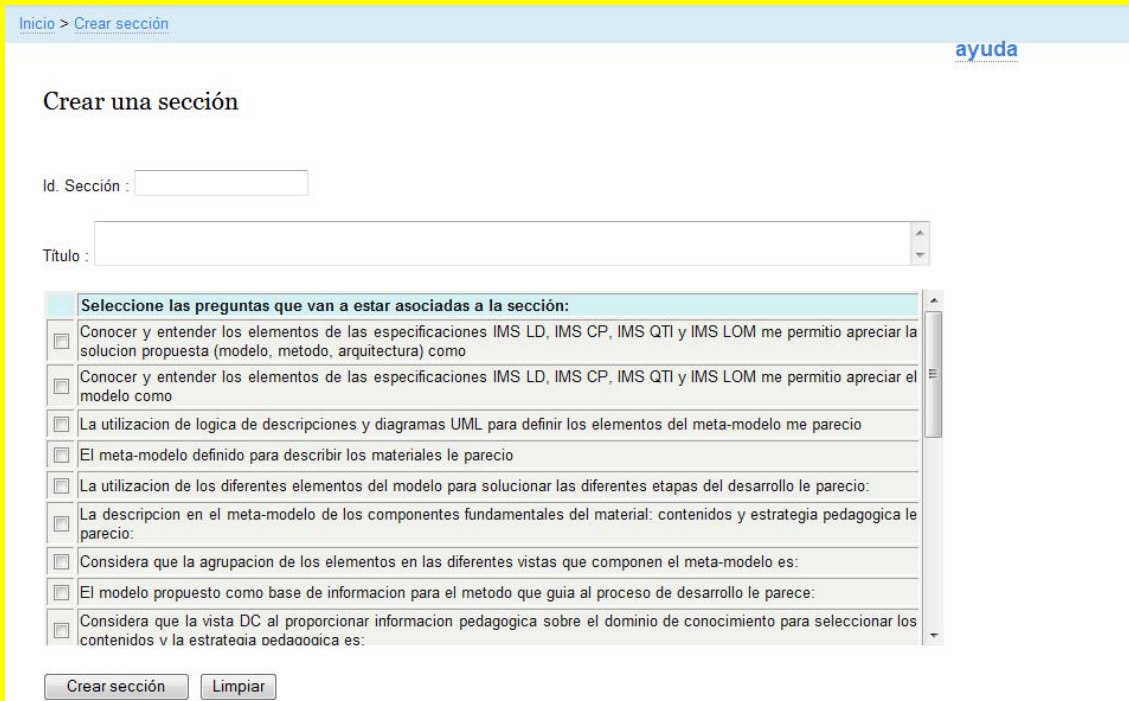
Figura 74. Página para eliminar un cuestionario

3) Hacer *click* en “Eliminar cuestionario” y si el cuestionario se ha eliminado correctamente de la base de datos, la aplicación mostrará un mensaje de éxito.

6.5.3. Gestión de secciones

Crear una sección:

- 1) Dentro del apartado “Secciones” elegir la opción “Crear sección” y se cargará una interfaz como la de la Figura 75.
- 2) Se deben rellenar los campos con los datos pertinentes y a continuación pulsar en “Crear sección”.
- 3) Para finalizar, la aplicación muestra un mensaje de éxito de la operación que contiene los datos de la sección creada.



Inicio > Crear sección

ayuda

Crear una sección

Id. Sección :

Título :

Seleccione las preguntas que van a estar asociadas a la sección:

- ☐ Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar la solución propuesta (modelo, metodo, arquitectura) como
- ☐ Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar el modelo como
- ☐ La utilizacion de logica de descripciones y diagramas UML para definir los elementos del meta-modelo me parecio
- ☐ El meta-modelo definido para describir los materiales le parecio
- ☐ La utilizacion de los diferentes elementos del modelo para solucionar las diferentes etapas del desarrollo le parecio:
- ☐ La descripcion en el meta-modelo de los componentes fundamentales del material: contenidos y estrategia pedagogica le parecio:
- ☐ Considera que la agrupacion de los elementos en las diferentes vistas que componen el meta-modelo es:
- ☐ El modelo propuesto como base de informacion para el metodo que guia al proceso de desarrollo le parece:
- ☐ Considera que la vista DC al proporcionar informacion pedagogica sobre el dominio de conocimiento para seleccionar los contenidos y la estrategia pedagogica es:

Figura 75. Página para crear una sección

Consultar los datos de una sección:

- 1) Elegir la opción "Consultar sección" del apartado "Secciones" del menú desplegable.
- 2) Elegir la sección por su título y click en "Mostrar información de la sección".

La Figuras 76 muestra un ejemplo de consulta de información de una sección.

ayuda

Consultar los datos de una sección

Sección: Sobre el meta-modelo propuesto

Id. Sección: **seccion02**

Título: **Sobre el meta-modelo propuesto**

Preguntas
Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar el modelo como
La utilizacion de logica de descripciones y diagramas UML para definir los elementos del meta-modelo me parecio
El meta-modelo definido para describir los materiales le parecio
La utilizacion de los diferentes elementos del modelo para solucionar las diferentes etapas del desarrollo le parecio:
La descripcion en el meta-modelo de los componentes fundamentales del material: contenidos y estrategia pedagogica le parecio:
Considera que la agrupacion de los elementos en las diferentes vistas que componen el meta-modelo es:
El modelo propuesto como base de informacion para el metodo que guia al proceso de desarrollo le parece:
Considera que la vista DC al proporcionar informacion pedagogica sobre el dominio de conocimiento para seleccionar los contenidos y la estrategia pedagogica es:
Considera que la vista P al facilitar informacion pedagogica sobre una situacion instructiva especifica es:
Considera que la vista S como contenedora de las relaciones snetre las descripciones del material y las de caracter mas tecnico de los estandares es:

Figura 76. Página para consultar los datos de una sección

Modificar los datos de una sección:

- 1) Elegir la opción "Modificar sección" del apartado "Secciones" del menú desplegable.
- 2) Elegir la sección por su título y *click* en "Mostrar información de la sección".

Esta funcionalidad permite modificar el título de la sección y las preguntas que contiene. Se pueden añadir o eliminar preguntas. En caso de añadir preguntas, permite tanto añadir preguntas previamente existentes en la base de datos pero también desde esta misma funcionalidad se puede crear una pregunta nueva que se asociará directamente a la sección seleccionada.

Las Figuras 77, 78 y 79 muestran un ejemplo de secuencia de pasos a seguir para modificar una sección:



Home Usuarios Cuestionarios Secciones Preguntas Estadísticas

Inicio > Modificar sección [ayuda](#)

Modificar los datos de una sección

Sección: Conocimientos sobre estándares educativos

Mostrar información de la sección

Id. Sección: seccion01

Título: Conocimientos sobre estándares educativos

Preguntas

Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar la solucion propuesta (modelo, metodo, arquitectura) como

Guardar cambios Limpiar

Añadir preguntas a la sección

Eliminar preguntas de la sección

Figura 77. Página para elegir la sección a modificar

Home Usuarios Cuestionarios Secciones Preguntas Estadísticas [ayuda](#)

Inicio > Modificar sección > Añadir Preguntas

Modificar Sección - Añadir preguntas

Id. Sección: seccion01

Título: Conocimientos sobre estándares educativos

Preguntas

Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar la solucion propuesta (modelo, metodo, arquitectura) como

Guardar cambios Limpiar

Añadir preguntas nuevas

Añadir preguntas existentes

[< VOLVER](#)

Figura 78. Página para modificar los datos de una sección

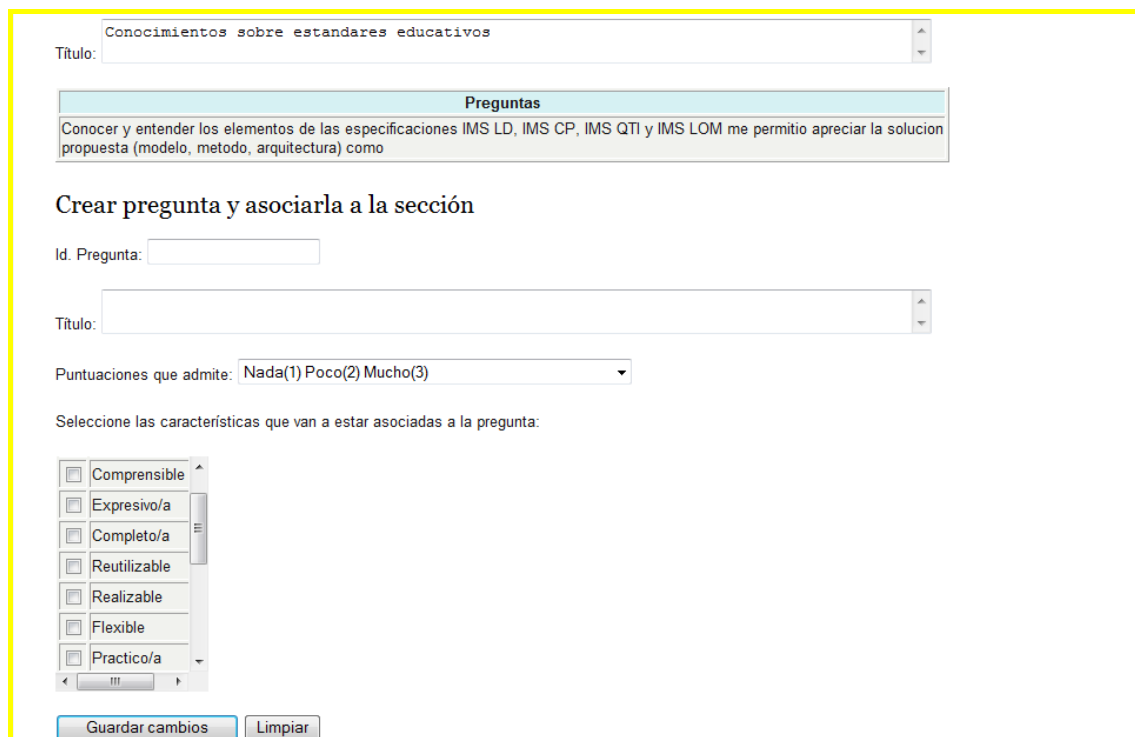


Figura 79. Página para añadir una pregunta nueva a la sección seleccionada

Eliminar una sección:

- 1) Elegir la opción "Eliminar sección" del apartado "Secciones" del menú desplegable.
- 2) Elegir la sección por su título y *click* en "Mostrar información de la sección".



Figura 80. Página para eliminar una sección

3) Hacer *click* en “Eliminar” y si la sección ha sido eliminada correctamente de la base de datos, la aplicación mostrará un mensaje de éxito.

6.5.4. Gestión de preguntas

Crear una pregunta:

- 1) Dentro del apartado “Preguntas” elegir la opción “Crear pregunta” y se cargará una interfaz como la de la Figura 81.
- 2) Se deben rellenar los campos con los datos pertinentes y a continuación pulsar en “Crear pregunta”.
- 3) Para finalizar, la aplicación muestra un mensaje de éxito de la operación que contiene los datos de la pregunta creada.

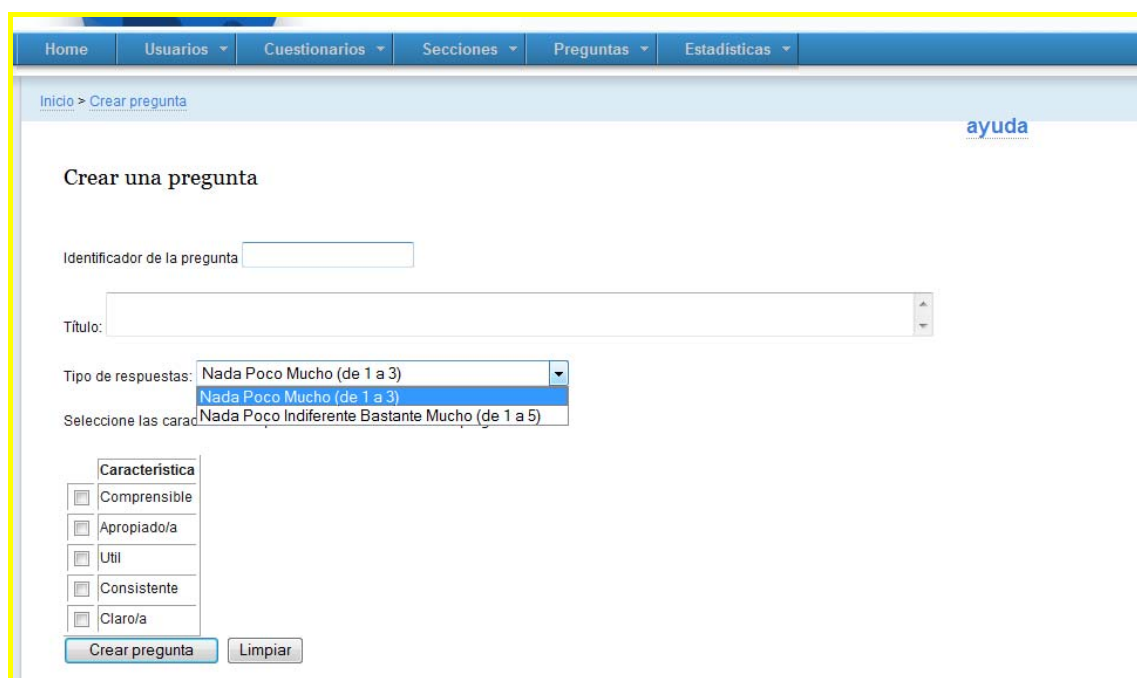


Figura 81. Página para crear una pregunta

Consultar los datos de una pregunta:

- 1) Elegir la opción “Consultar pregunta” del apartado “Preguntas” del menú desplegable.
- 2) Elegir la pregunta por su título y *click* en “Mostrar datos pregunta”.

La Figura 82 muestra un ejemplo de consulta de información de una pregunta.

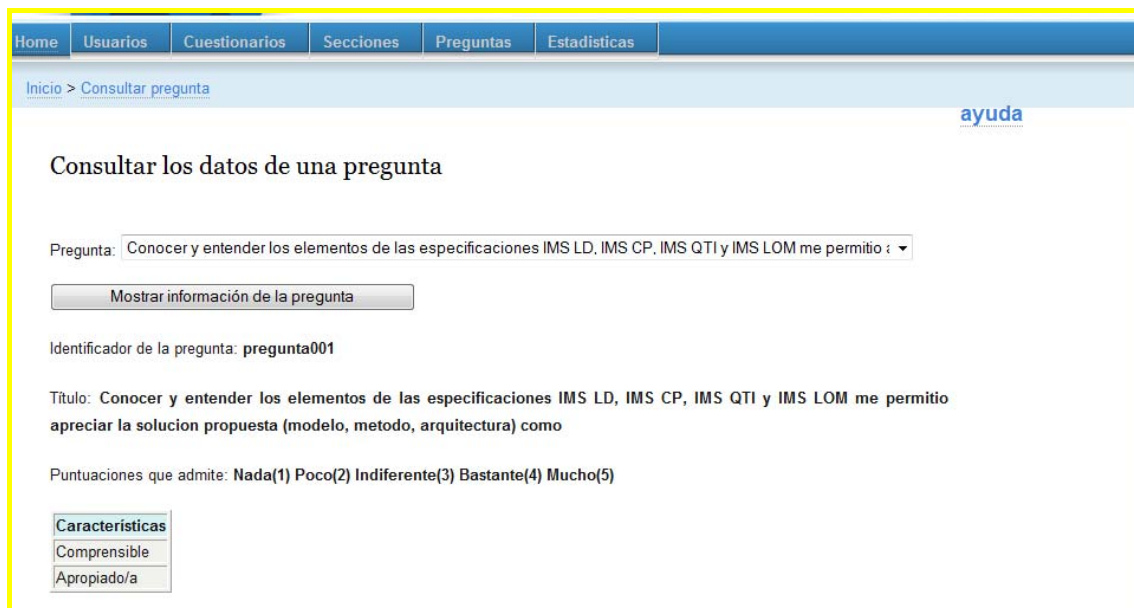


Figura 82. Página para consultar los datos de una pregunta

Modificar los datos de una pregunta:

- 1) Elegir la opción "Modificar pregunta" del apartado "Preguntas" del menú desplegable.
- 2) Elegir la pregunta por su título y *click* sobre "Mostrar información de la pregunta".

Esta funcionalidad permite modificar el título de la pregunta, el tipo de puntuación que permite y las preguntas que contiene. Se pueden añadir o eliminar características.

Las Figuras 83 y 84 muestran un ejemplo de secuencia de pasos a seguir para modificar una pregunta:

Modificar los datos de una pregunta

Pregunta: Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio : ▾

[Mostrar información de la pregunta](#)

Id. Pregunta: pregunta001

Título: Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar la solucion propuesta (modelo, ▾

Puntuaciones que admite: Nada(1) Poco(2) Indiferente(3) Bastante(4) Mucho(5) ▾

Seleccione las características que van a estar asociadas a la pregunta:

☐ Reutilizable
☐ Realizable
☒ Flexible
☐ Practico/a
☒ Apropiado/a
☐ Util
☐ Consistente

[Guardar cambios](#) [Limpiar](#)

Figura 83. Página para modificar los datos de una pregunta

admin [Cerrar sesión](#)

[Home](#) [Usuarios](#) [Cuestionarios](#) [Secciones](#) [Preguntas](#) [Estadísticas](#)

[Inicio](#) > [Modificar pregunta](#) [ayuda](#)

La pregunta ha sido modificada con éxito:

Id. Pregunta: pregunta001

Título: Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar la solucion propuesta (modelo, metodo, arquitectura) como

Puntuaciones que admite: Nada(1) Poco(2) Indiferente(3) Bastante(4) Mucho(5)

Características a valorar:

1) Comprensible
 2) Flexible
 3) Apropiado/a

Figura 84. Página que muestra que la pregunta se ha modificado con éxito

Eliminar una pregunta:

- 1) Elegir la opción "Eliminar pregunta" del apartado "Preguntas" del menú desplegable.



- 2) Elegir la pregunta por su título y *click* en “Mostrar información de la pregunta”.
- 3) Hacer *click* en “Eliminar” y si la pregunta se ha eliminado correctamente de la base de datos, la aplicación mostrará un mensaje de éxito.

Figura 85. Página para eliminar una pregunta

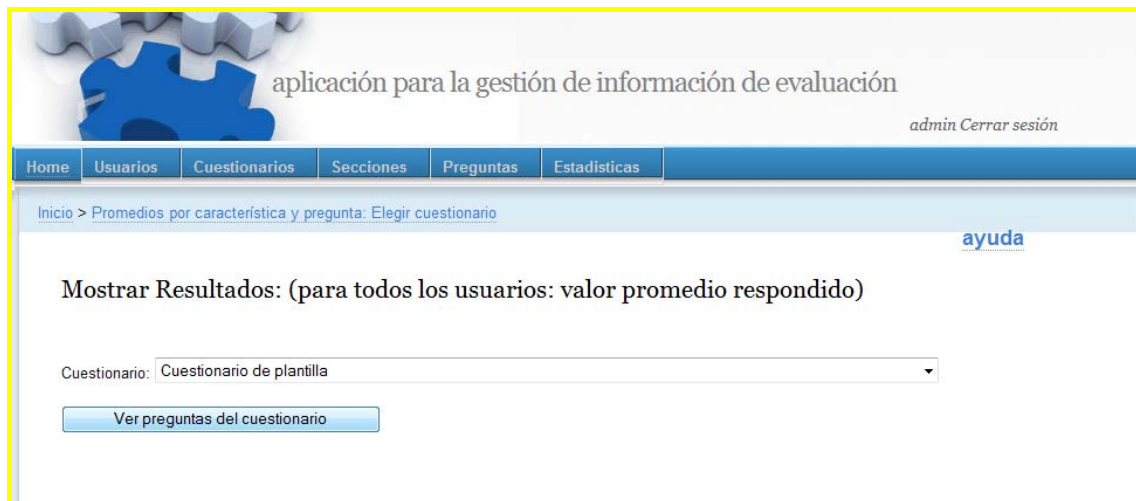
6.5.5. Estadísticas

Para un cuestionario, mostrar la puntuación media por pregunta característica:

- 1) Elegir la opción “Promedios de puntuaciones de un cuestionario” del menú desplegable del apartado “Estadísticas”.
- 2) Hacer *click* en “Ver preguntas del cuestionario”. Mostrará las preguntas que contiene el cuestionario.
- 3) A continuación hacer *click* en “Mostrar resultados”. La aplicación mostrará una pregunta por página. Cada pregunta a su vez contiene una serie de características a valorar, de las cuales se muestra el promedio de las puntuaciones que los usuarios han otorgado a esa característica de la pregunta.

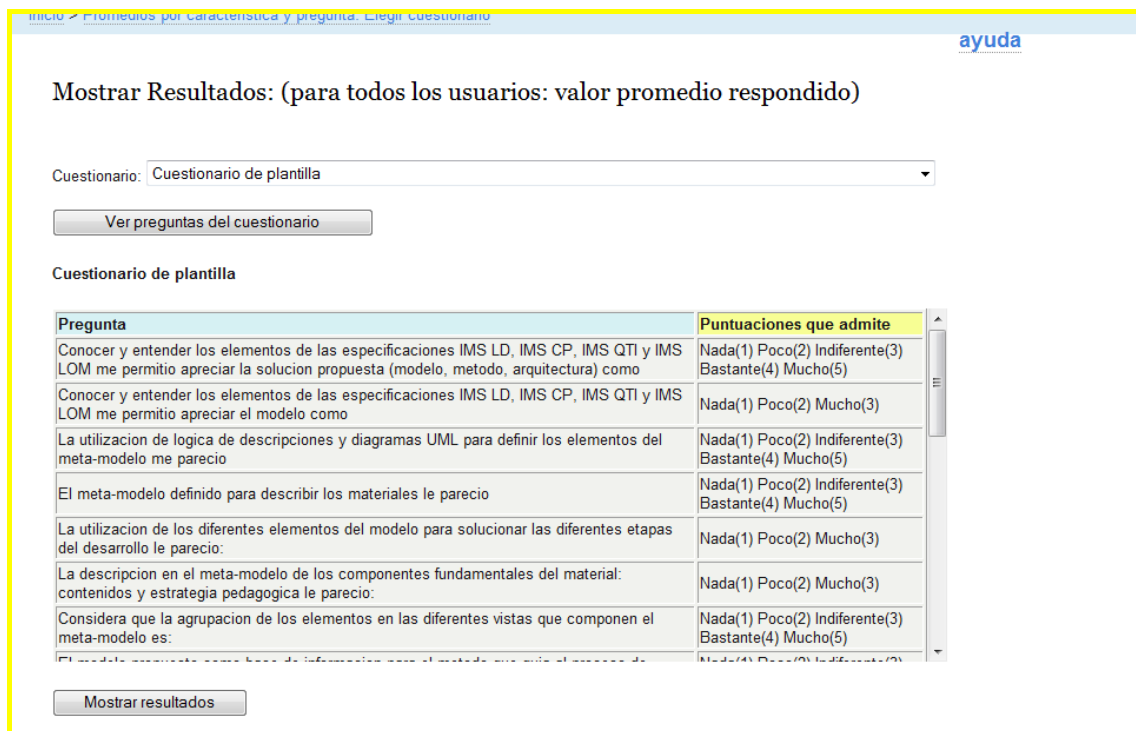
4) Para ir a la siguiente pregunta, hacer *click* en “Siguiente pregunta”. Y así hasta llegar al fin del cuestionario, en cuyo caso, la aplicación avisará al usuario.

Las Figuras 86, 87 88 y 89 reflejan la secuencia de pasos a seguir:



The screenshot shows the web application interface. At the top, there is a header with the title 'aplicación para la gestión de información de evaluación' and a user link 'admin Cerrar sesión'. Below the header is a navigation bar with tabs: Home, Usuarios, Cuestionarios, Secciones, Preguntas, and Estadísticas. The main content area has a breadcrumb trail: 'Inicio > Promedios por característica y pregunta: Elegir cuestionario'. There is a 'ayuda' link in the top right. The main heading is 'Mostrar Resultados: (para todos los usuarios: valor promedio respondido)'. Below this is a dropdown menu for 'Cuestionario:' with 'Cuestionario de plantilla' selected. A button 'Ver preguntas del cuestionario' is located below the dropdown.

Figura 86. Página para seleccionar el cuestionario del que se desean ver sus promedios



The screenshot shows the web application interface displaying the selected questionnaire questions. The breadcrumb trail is 'Inicio > Promedios por característica y pregunta: Elegir cuestionario'. The 'ayuda' link is in the top right. The main heading is 'Mostrar Resultados: (para todos los usuarios: valor promedio respondido)'. Below this is a dropdown menu for 'Cuestionario:' with 'Cuestionario de plantilla' selected. A button 'Ver preguntas del cuestionario' is located below the dropdown. Below this, the section 'Cuestionario de plantilla' contains a table with the following data:

Pregunta	Puntuaciones que admite
Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar la solucion propuesta (modelo, metodo, arquitectura) como	Nada(1) Poco(2) Indiferente(3) Bastante(4) Mucho(5)
Conocer y entender los elementos de las especificaciones IMS LD, IMS CP, IMS QTI y IMS LOM me permitio apreciar el modelo como	Nada(1) Poco(2) Mucho(3)
La utilizacion de logica de descripciones y diagramas UML para definir los elementos del meta-modelo me parecio	Nada(1) Poco(2) Indiferente(3) Bastante(4) Mucho(5)
El meta-modelo definido para describir los materiales le parecio	Nada(1) Poco(2) Indiferente(3) Bastante(4) Mucho(5)
La utilizacion de los diferentes elementos del modelo para solucionar las diferentes etapas del desarrollo le parecio:	Nada(1) Poco(2) Mucho(3)
La descripcion en el meta-modelo de los componentes fundamentales del material: contenidos y estrategia pedagogica le parecio:	Nada(1) Poco(2) Mucho(3)
Considera que la agrupacion de los elementos en las diferentes vistas que componen el meta-modelo es:	Nada(1) Poco(2) Indiferente(3) Bastante(4) Mucho(5)
El modelo presentado como base de informacion para el estado que es el proceso de	Nada(1) Poco(2) Indiferente(3) Bastante(4) Mucho(5)

At the bottom of the table, there is a button 'Mostrar resultados'.

Figura 87. Página que muestra las preguntas del cuestionario seleccionado

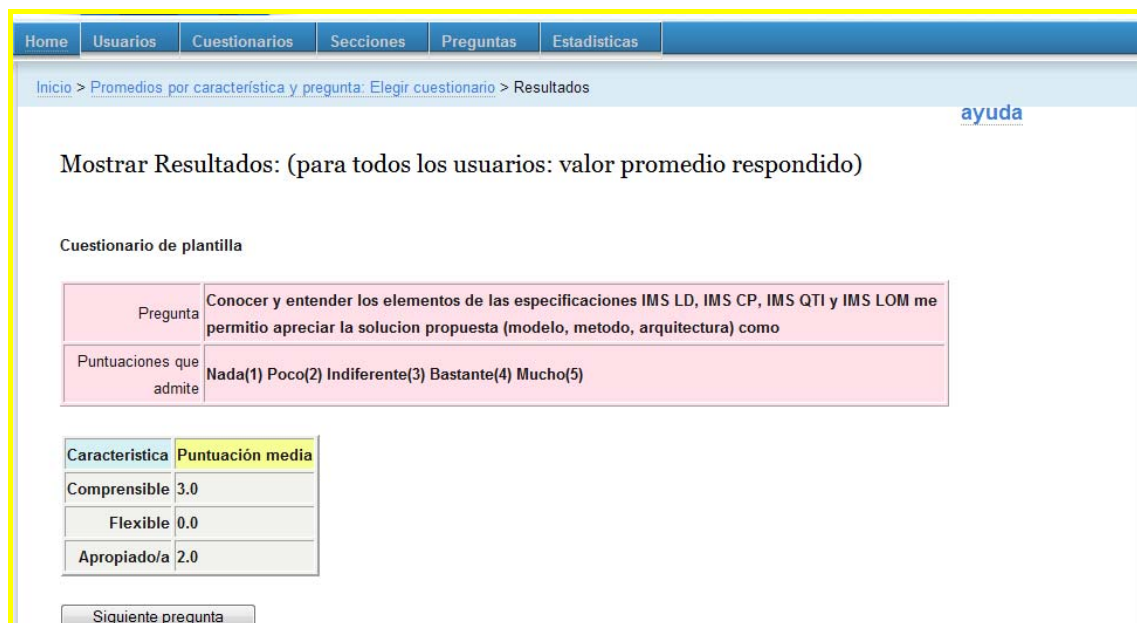


Figura 88. Página que muestra las puntuaciones medias por característica y preguntas

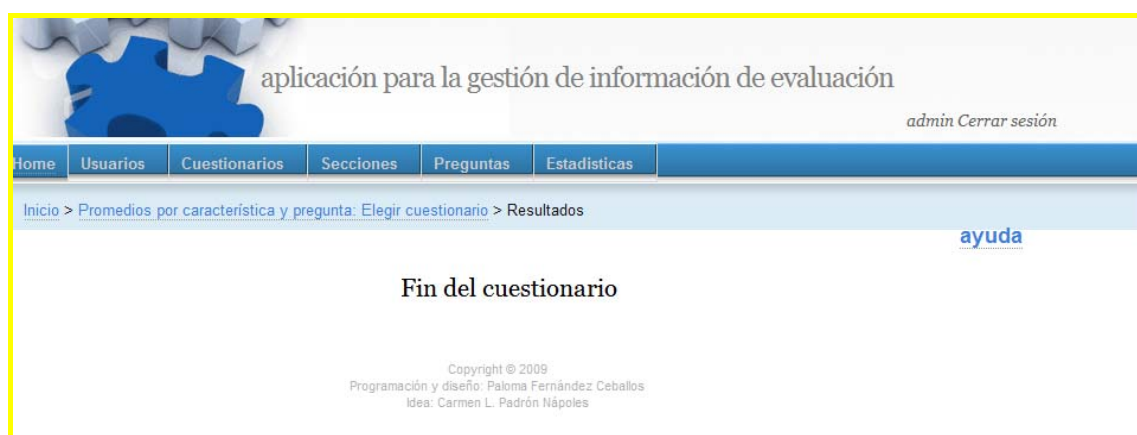


Figura 89. Página que indica que el cuestionario ha llegado a su fin

Para un cuestionario, mostrar el porcentaje de usuarios que han respondido cada opción por pregunta y característica:

- 1) Elegir la opción "Porcentaje de usuarios que han respondido cada opción" del menú desplegable del apartado "Estadísticas".
- 2) Hacer *click* en "Ver preguntas del cuestionario". Mostrará las preguntas que contiene el cuestionario.
- 3) A continuación hacer *click* en "Mostrar resultados". La aplicación mostrará una pregunta por página. Cada pregunta a su vez contiene una serie de características a valorar, de las cuales se muestra el porcentaje de

usuarios que ha respondido cada opción de cada característica de la pregunta.

4) Para ir a la siguiente pregunta, hacer *click* en “Siguiente pregunta”. Y así hasta llegar al fin del cuestionario, en cuyo caso, la aplicación avisará al usuario.

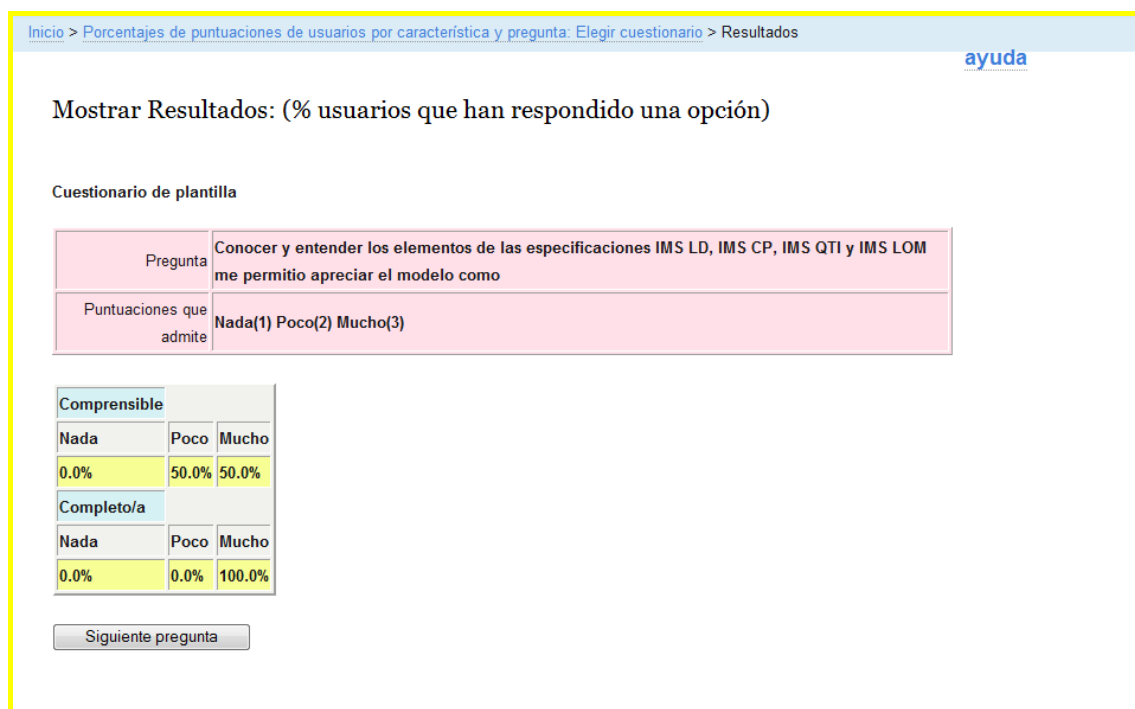


Figura 90. Página que muestra el porcentaje de usuarios que ha respondido cada opción

7. GESTIÓN DEL PROYECTO

El ciclo de vida que ha seguido el proyecto es el de modelo en cascada (Figura 91), de tal modo que el inicio de cada etapa deba esperar a la finalización de la inmediatamente anterior. De esta forma, cualquier error de diseño detectado en la etapa de prueba conduce necesariamente al rediseño y nueva programación del código afectado, aumentando los costes del desarrollo.

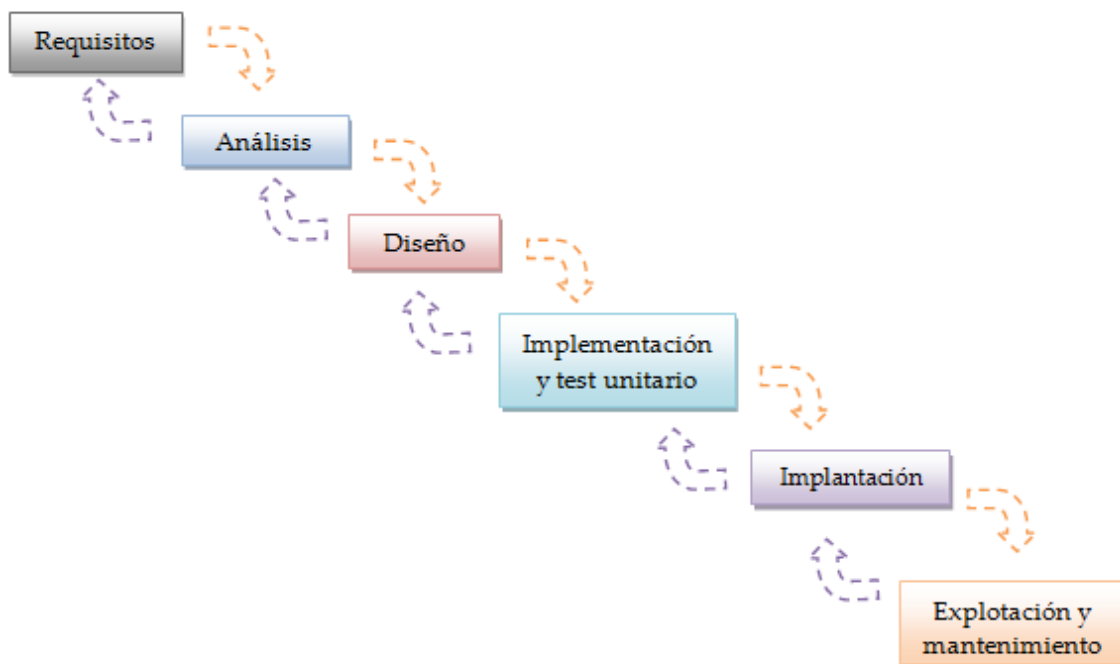


Figura 91. Ciclo de vida en cascada

7.1. Planificación del proyecto

El proyecto se inició a finales de Junio de 2008 y concluyó a finales de Junio de 2009, encontrándose mayores dificultades en las tareas de diseño e implementación de la aplicación. Concretamente con el diseño de la base de datos que hubo que cambiarlo varias veces a lo largo del proyecto y con la implementación de las funcionalidades de responder un cuestionario y de visualización de estadísticas.

En la Figura 92 queda reflejada la planificación realizada para el proyecto, mostrando con detalle la división de tareas, la duración de cada tarea, así como las fechas de inicio y fin planteadas para cada una de ellas.

Esta planificación no incluye ninguna gestión de riesgos dado el carácter académico del presente proyecto.





























		Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1		Recogida de requisitos de usuario	23 días	mié 25/06/08	vie 25/07/08	
2		- Análisis	49 días	lun 28/07/08	jue 02/10/08	1
3		Análisis de requisitos de usuario	13 días	lun 28/07/08	mié 13/08/08	
4		Definición de los casos de uso	7 días	jue 14/08/08	vie 22/08/08	3
5		Casos de uso extendidos	5 días	lun 15/09/08	vie 19/09/08	4
6		Diagramas de actividad	7 días	lun 22/09/08	mar 30/09/08	5
7		Diagrama de clases	2 días	mié 01/10/08	jue 02/10/08	6
8		Fin de análisis	0 días	jue 02/10/08	jue 02/10/08	7
9		- Diseño	37 días	vie 03/10/08	lun 24/11/08	8
10		Definición de los requisitos de software	5 días	vie 03/10/08	jue 09/10/08	
11		Diseño de la base de datos	7 días	vie 10/10/08	lun 20/10/08	10
12		- Diseño de la lógica de negocio	15 días	mar 21/10/08	lun 10/11/08	11
13		Diagrama de clases detallado	5 días	mar 21/10/08	lun 27/10/08	
14		Diagramas de secuencia	10 días	mar 28/10/08	lun 10/11/08	13
15		Diseño de la interfaz de usuario	10 días	mar 11/11/08	lun 24/11/08	12
16		Fin de diseño	0 días	lun 24/11/08	lun 24/11/08	15
17		- Implementación	118 días	mar 25/11/08	vie 08/05/09	16
18		Script de la base de datos	4 días	mar 25/11/08	vie 28/11/08	
19		Módulo Gestión de usuarios	13 días	lun 01/12/08	mié 17/12/08	18
20		Módulo Gestión de preguntas	15 días	jue 18/12/08	mié 07/01/09	19
21		Módulo Gestión de secciones	15 días	jue 08/01/09	mié 28/01/09	20
22		Módulo Gestión de cuestionarios	10 días	jue 29/01/09	mié 11/02/09	21
23		Responder cuestionario	7 días	jue 12/02/09	vie 20/02/09	22
24		Visualización de estadísticas	15 días	lun 23/02/09	vie 13/03/09	23
25		Implementación de las vistas	20 días	lun 16/03/09	vie 10/04/09	24
26		Integración (modelo + vista + controlador)	18 días	lun 13/04/09	mié 06/05/09	25
27		Fin de la implementación	0 días	vie 08/05/09	vie 08/05/09	26
28		- Pruebas	124 días	mar 02/12/08	vie 22/05/09	18
29		Pruebas unitarias con Junit	75 días	mar 02/12/08	lun 16/03/09	
30		Pruebas de aceptación	12 días	jue 07/05/09	vie 22/05/09	
31		Implantación	2 días	lun 25/05/09	mar 26/05/09	28
32		Documentación del proyecto	36 días	vie 08/05/09	vie 26/06/09	27
33		Fin de proyecto	0 días	vie 26/06/09	vie 26/06/09	32

Figura 92. Planificación de tareas del proyecto

La planificación recoge siete tareas principales:

- 1. Recogida de requisitos de usuario:** Durante esta primera fase se mantuvieron varias reuniones con la tutora del proyecto, Carmen Padrón, que en este caso actuaba de cliente. De estas reuniones se



fueron extrayendo los requisitos necesarios para cumplir con la funcionalidad de la aplicación final.

- 2. Análisis:** Esta tarea comprende todo lo referente a la fase de análisis de la aplicación. Una vez recogidos los requisitos de usuario, se ha hecho un análisis de los mismos, a partir de los cuales se definirían los casos de uso, que son los que esquematizarían el funcionamiento que tendría la aplicación.
- 3. Diseño:** La tarea de diseño comprende la fase previa a la implementación de la aplicación, por lo que tiene que quedar claro para el programador que es lo que tiene que hacer y cómo. Para ello se han realizado una serie de diagramas que reflejan el diseño detallado que debe seguir la aplicación. La primera tarea ha sido definir los requisitos de software, a continuación se planteó el diseño de la base de datos, exponiendo así el modelo relacional de la misma. Se continuó con el diseño de la lógica de negocio, extrayendo así el diagrama de clases detallado. Y por último se ha llevado a cabo el diseño de la interfaz.
- 4. Implementación:** Esta tarea comprende toda la codificación de la aplicación. En primer lugar se creó el script de la base de datos para tener los datos físicos con los que se iba a trabajar. A continuación se implementaron cada una de las funcionalidades separadas en *módulos* de gestión. Éstas aparecen detalladas en la Figura 92.
- 5. Pruebas:** La fase de pruebas la componen dos tipos de prueba diferentes: las unitarias, que se han ido haciendo en paralelo a la fase de implementación. Se han ido probando cada una de las funcionalidades por separado con la herramienta JUnit. Una vez completada la fase de implementación, se han llevado a cabo las pruebas de aceptación de la aplicación, que verifican el funcionamiento de la aplicación al completo.
- 6. Implantación:** Esta fase comprende las acciones necesarias para poner en funcionamiento la aplicación en un entorno de producción. Para ello se ha configurado el contenedor web de la aplicación así como el gestor de base de datos para la conexión con la base de datos.
- 7. Documentación:** Para finalizar, se ha llevado a cabo la fase de documentación en la que queda reflejado todo el trabajo realizado durante el ciclo de vida del proyecto. Además, los tres primeros capítulos introducen al usuario al concepto de la usabilidad, explican



el por qué de los cuestionarios de evaluación y hacen referencia a la arquitectura y tecnologías empleadas.

La Figura 93 muestra el Diagrama de Gantt resultante de la planificación realizada. En él se pueden apreciar las tareas descritas anteriormente, que se han ido llevando a cabo a lo largo del tiempo de duración del proyecto.

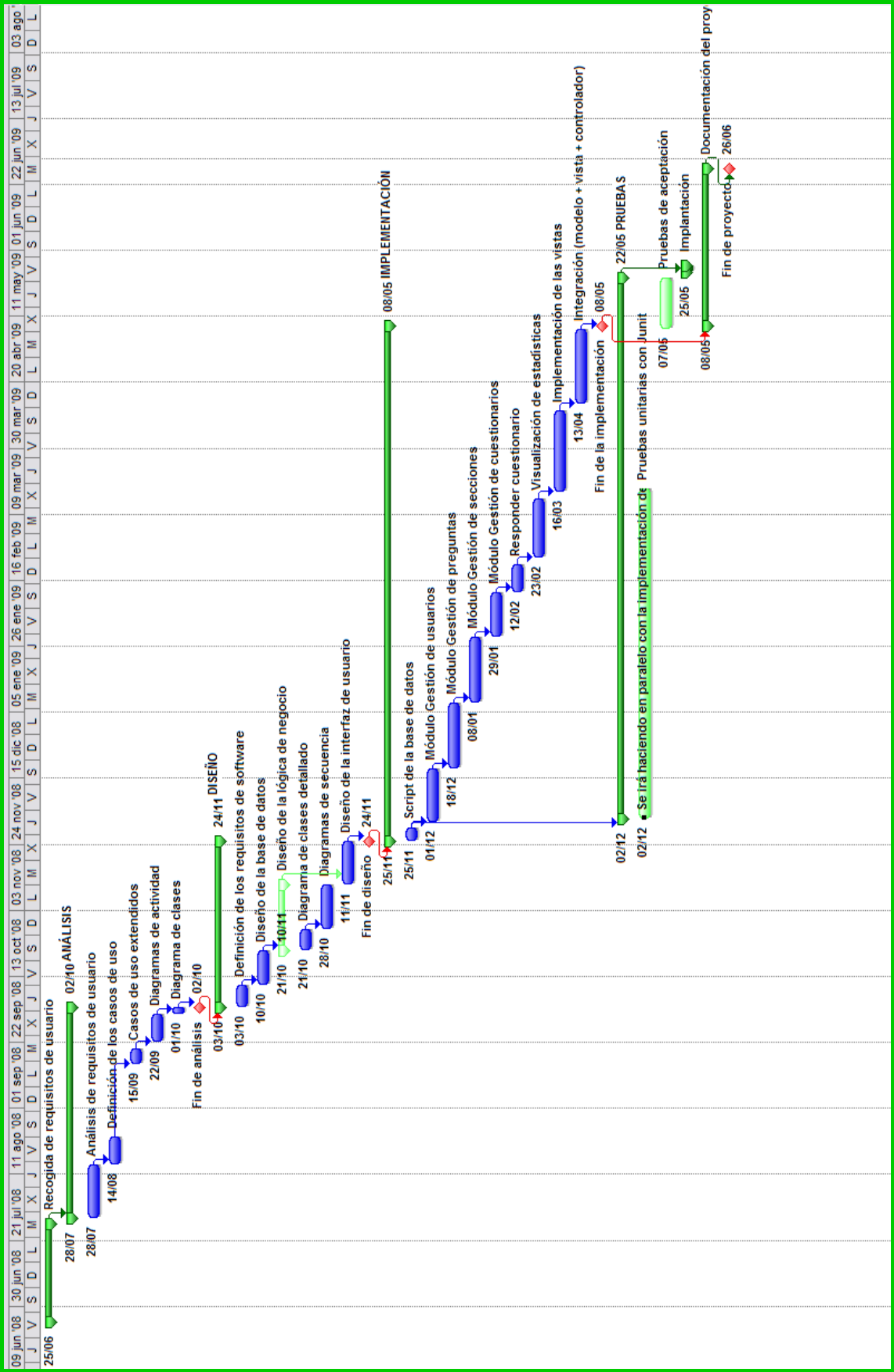


Figura 93. Diagrama de Gantt



7.2. Análisis de costes

Los costes que conlleva el desarrollo del presente proyecto se han dividido de la siguiente forma:

1. Personal implicado
2. Material e infraestructuras utilizadas
3. Otros

7.2.1. Personal

Para el desarrollo del proyecto se han necesitado profesionales de tres perfiles distintos pero complementarios, entre ellos un analista funcional, un analista orgánico y un analista/programador. Debido al carácter académico del proyecto, éste ha sido llevado a cabo por una única persona que ha hecho las veces de analista y programador, por lo que se ha prescindido del jefe de proyecto.

Por tanto, para calcular el coste de personal se han tenido en cuenta dos perfiles: el analista funcional (que ha hecho las veces de analista orgánico) y el analista/programador.

La tabla 40 refleja el número de días que ha empleado cada uno de estos perfiles en el desarrollo del proyecto. Este número de días de trabajo multiplicado por el coste diario de cada profesional, da como resultado el coste total de personal.

Tarea / Nº días	Analista funcional/orgánico	Analista-Programador
Recogida de requisitos de usuario	23	0
Análisis		
Análisis de requisitos de usuario	13	0
Definición de los casos de uso	7	0
Casos de uso extendidos	5	0
Diagramas de actividad	7	0
Diagrama de clases de análisis	2	0
Diseño		
Definición de los requisitos de software	5	1
Diseño de la base de datos	7	2
Diseño de la lógica de negocio		
Diagrama de clases detallado	5	3



Diagramas de secuencia	10	8
Diseño de la interfaz de usuario	10	5
Implementación		
Script de la base de datos	2	4
Módulo Gestión de usuarios	0	13
Módulo Gestión de preguntas	0	15
Módulo Gestión de secciones	0	15
Módulo Gestión de cuestionarios	0	10
Responder cuestionario	0	7
Visualización de estadísticas	0	15
Implementación de las vistas	0	20
Integración (M + V + C)	0	18
Pruebas		
Pruebas unitarias con JUnit	0	75*
Pruebas de aceptación	12	12
Implantación	0	2
Documentación del proyecto		
TOTAL:	108	150

Tabla 40. Costes del personal implicado en el proyecto

* Los setenta y cinco días de las pruebas unitarias no suman ya que van solapadas con la implementación de los módulos.

En la siguiente tabla se muestra el resumen de días y costes totales:

	Días trabajados	€ / hora	Coste Total
Analista Funcional/Orgánico	108 → 864 h	15	12.960 €
Analista - Programador	150 → 1200 h	10	12.000 €
		TOTAL:	24.960 €

Tabla 41. Resumen de días y costes totales del personal

7.2.2. Material e infraestructuras

Para el desarrollo del proyecto ha sido necesario el uso de material, tanto hardware como software, así como material de oficina empleado para la toma de notas auxiliares y documentación final o la conexión a internet, imprescindible para poder estar en contacto con el cliente, y poderle mandar las versiones pertinentes del proyecto en cada fase del mismo.



Como resultado queda la siguiente tabla que indica los costes de material desglosados:

	Precio	Cantidad	Subtotal
Portátil Packard Bell EasyNote CORE 2 DUO T7200 2.00 Ghz 2 G Ram	1.099	1	1.099
Licencia de Microsoft Windows XP	132	1	132
CD-R 80 min	0,50	3	1,50
Papel A5 (500 hojas)	3	2	6
Tinta impresora	30	1	30
Conexión Internet	30	4	120
			1388,5 €
		IVA 16%	222,16 €
		TOTAL:	1.610,66 €

Tabla 42. Costes de materiales e infraestructuras

7.2.3. Otros

En esta sección se muestra la tabla con los costes que no entrarían ni en personal ni en material empleado. Se ha tenido en cuenta el coste de transporte de Getafe a Leganés empleado para las reuniones con el cliente y otras actividades relacionadas con el desarrollo del proyecto.

	Precio	Cantidad	Subtotal
Bono Transporte 10 viajes (Getafe - Leganés)	6,70	7	46,9
		TOTAL:	46,9 €

Tabla 43. Otros gastos

7.2.4. Resumen de costes

La tabla 44 resume el balance del coste total del proyecto, para ello se suman los costes obtenidos anteriormente.

	Costes
Personal	24.960
Material e Infraestructuras	1.610,66
Otros	46,9
TOTAL:	26.617,56 €

Tabla 44. Resumen de costes del proyecto

A la cantidad anterior hay que sumarle el 20% de beneficios:

$$26.617,56 + 5.323,51 = 31.941,07 \text{ €}$$

El proyecto tiene un coste total de **31.941,07 €** (Treinta y un mil novecientos cuarenta y un euros con siete céntimos).



8. CONCLUSIONES

Como capítulo final del documento, se presentan las conclusiones extraídas a lo largo de este Proyecto Fin de Carrera, así como las líneas futuras que propone.

8.1. Conclusiones finales

El proyecto llevado a cabo se centra en la recogida de datos por medio de cuestionarios de evaluación. El uso de estos cuestionarios es bueno para la recogida de respuestas a preguntas específicas, y su mayor ventaja es que, con pocos recursos, lo pueden responder un grupo muy amplio de usuarios.

Se ha pretendido implementar un diseño de cuestionario lo más general posible que permita generar cuestionarios para casi cualquier tipo de sistema, así con una sola aplicación podremos evaluar sistemas de todo tipo. Esto hará que no haga falta solicitar a una entidad determinada un tipo de cuestionario específico para un sistema en concreto, lo que hará que se ahorre en tiempo y dinero y se ganará en eficiencia y eficacia.

Para el desarrollo de proyecto se ha implementado el patrón Modelo Vista Controlador, arquitectura que separa claramente la lógica de negocio de la presentación. A continuación se enumeran algunas de las ventajas que supone su aplicación:

- ✚ Sencillez para crear distintas representaciones de los mismos datos.
- ✚ Facilidad para la realización de pruebas unitarias de los componentes, así como de aplicar desarrollo guiado por pruebas (TDD).
- ✚ Reutilización de los componentes.
- ✚ Simplicidad en el mantenimiento de los sistemas.
- ✚ Facilidad para desarrollar prototipos rápidos.
- ✚ Los desarrollos suelen ser más escalables.

Estas son algunas de las razones por las cuales es una arquitectura usada en grandes sistemas a nivel empresarial.

Eclipse [2] como entorno nos ha permitido implementar la aplicación en Java, por lo que se tiene un sistema multiplataforma. Además, para la aplicación del patrón MVC se ha utilizado Struts ya que es un excelente *framework* que ayuda a implementar esta arquitectura para aplicaciones web J2EE como es el caso.

La interfaz de la aplicación la definen las diferentes vistas de la arquitectura previamente definida. La interfaz se compone de un conjunto



de páginas HTML dinámicas, conseguidas con el uso de HTML + Javascript + DOM [1]. La ventaja de emplear DOM [1] es que permite a los programadores disponer de un control muy preciso sobre la estructura del documento HTML o XML que están manipulando. DOM permite añadir, eliminar, modificar y reemplazar cualquier nodo de cualquier documento de forma sencilla.

Además el proyecto se ha realizado tomando uso de las APIs y herramientas *Open Source*, por lo que en lo que a software se refiere se ha invertido lo mínimo.

Una vez concluido el proyecto vemos la enorme importancia que supone una alta usabilidad en las aplicaciones. Para aumentar el rendimiento de cualquier sistema es muy importante saberlo usar y así poderle sacar un mayor provecho. Muchas veces queremos hacer aplicaciones tan completas que se van añadiendo funcionalidades, hasta llegar a un punto que se hace de una aplicación completa, una aplicación compleja. Además, se quiere hacer “tan bonito” que llega a ser muy difícil de usar para un usuario con conocimientos básicos en informática. Toda aplicación debe encontrar un equilibrio entre la completitud de la misma y la facilidad de su uso. Ya hemos visto a lo largo de este proyecto que los cuestionarios son una parte fundamental en la evaluación de la usabilidad de cualquier sistema.

Un interrogante que dejo en el aire es: ¿qué es mejor?, ¿El uso de cuestionarios generales o específicos? Los cuestionarios específicos veíamos que pueden aportar resultados más concretos al sistema que atañen, pero hay que pedirlos a una entidad en concreto, con el conveniente coste tanto en tiempo de realización como en euros que supone. Los cuestionarios generales permiten evaluar cualquier sistema aunque con menos precisión, pero siempre se pueden intentar “exclusivizar” para el sistema en concreto a evaluar. Debido a los inconvenientes de generalización que podrían acarrear este tipo de cuestionarios, se ha llevado a cabo esta aplicación web de gestión de cuestionarios para la evaluación de cualquier sistema, permitiendo de una manera más sencilla y mucho menos costosa el hecho de crear cuestionarios de evaluación para sistemas específicos.

En cuanto a aspectos algo más personales, decir que este proyecto me ha sido realmente útil para profundizar en conocimientos relacionados con la plataforma J2EE, el patrón MVC, Struts, Hibernate [3] así como el diseño de páginas web con HTML, Javascript, DOM [1] y CSS.

A estas alturas puedo decir que se han logrado los objetivos marcados lo que me provoca una enorme satisfacción tanto a nivel personal, porque sé que con dedicación y esfuerzo todo se supera, como a



nivel profesional, por todo el trabajo que ha supuesto la realización del proyecto durante todo este tiempo sin olvidarme, por supuesto, de todas y cada una de las asignaturas de la carrera, que, unas más que otras, todas han contribuido a que este proyecto sea hoy una realidad.

8.2. Trabajo futuro y ampliaciones

Este Proyecto Fin de Carrera queda abierto a posibles ampliaciones.

A continuación se citan algunas líneas de trabajo futuro que podrían tenerse en cuenta:

1. Lo referente a la asociación de secciones con preguntas y cuestionarios podría mejorarse utilizando AJAX (*Asynchronous JavaScript And XML*), es una técnica de desarrollo web que sirve para crear aplicaciones interactivas. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma sería posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significaría aumentar la interactividad, velocidad y usabilidad en las aplicaciones.
2. Para el usuario sería más útil y fácil de ver, el hecho de representar las estadísticas a través de gráficos. Podrían emplearse así gráficos que compararan resultados de varios cuestionarios a la vez. Lo que facilitaría la extracción de conclusiones.
3. La aplicación resultante contiene una *breadcrumb* (*miga de pan*) por vista. Ésta es estática, por lo que no representa la ruta completa que ha ido siguiendo un usuario para realizar una tarea. Se propone mejorar esta *breadcrumb* y hacerla dinámica, para facilitar el acceso a las vistas anteriores por las que se ha ido navegando para realizar dicha funcionalidad.
4. Sería conveniente hacer de este sistema uno multilingüe ya que generalizaría aún más la aplicación, aspecto que interesa si lo que queremos es obtener un sistema lo más flexible y eficaz posible.
5. Como última línea futura de trabajo se propone la posibilidad de editar el fichero de propiedades de Hibernate [3], para que pueda cambiarse la conexión a la base de datos, el usuario de acceso y la contraseña.



REFERENCIAS

- ACM SIGCHI. (1987), *Conference on Human Factors in Computing Systems*, Toronto.
- Almond, R. e. a. (2002), 'IMS question and test interoperability final specification 1.2. IMS global learning consortium'.
- Bias, R. G. (1994), *Usability Inspection Methods*, John Wiley and Sons, chapter The pluralistic walkthrough: coordinated empathies, pp. 63-76.
- Brajnik, G. (2000), Automated Web usability evaluation: Where is the limit?, in `6th Conference on Human Factor and the Web'.
- Brooke, J. (1996), *Usability Evaluation in Industry*, Taylor and Francis, chapter SUS: A 'quick and dirty' usability scale, pp. 189-194.
- Chin, J. P., Diehl, V. A. & Norman, K. L. (1988), Development of an instrument measuring user satisfaction of the human-computer interface, in `ACM Human Factors in Computing System Conference (CHI'88)', pp. 213-218.
- Davis, F. D. (1989), `Perceived usefulness, perceived ease of use, and user acceptance of information technology', *MIS Quarterly* 13(3), 227-247.
- Diaper, D. (1989), *Task Analysis in Human-Computer Interaction*, Ellis Horwood, chapter Task observation for human-computer interaction, pp. 210-237.
- Dix, A., Finlay, J., Abowd, G. & Beale, R. (1998), *Human-Computer Interaction*, Prentice-Hall Europe.
- Dumas, J. S. & Redish, J. C. (2000), *A practical guide to usability testing*, Intellect.
- Foody, W. (1999), *Constructing questions for interviews and questionnaires*, Cambridge University Press.
- Frokjaer, E., Hertzum, M. & Hornbeek, K. (2000), Measuring usability: are effectiveness, efficiency and satisfaction really correlated?, in `Conference on Human Factors in Computing Systems 2000', pp. 345-352.
- García, E., Sicilia, M. A., González, L. & Hilera, J. R. (2002), Machine learning techniques in usability-evaluation questionnaire systems, in 'Int. Conference on Machine Learning (Learning02)'.



- García, E., Sicilia, M. A., Hilera, J. R. & Gutiérrez, J. A. (2002), Extending quest and test learning technology specifications with enhanced questionnaire models, 'Intl. Conference on Information Technology Based Higher Education and Training (ITHET02)'.
- García, E., Sicilia, M. A., Hilera, J. R. & Gutiérrez, J. A. (2001), Extracting knowledge from usability evaluation databases, *in* '8th conf on Human-Computer Interaction (INTERACT 2001)', pp. 713-715.
- García, E., Sicilia, M., Hilera, J. & Gutiérrez, J. (2001), *Advances in Human-Computer Interaction*, Typorama, chapter Computer-Aided Usability Evaluation: A Questionnaire Case Study, pp. 85-91.
- García Barriocanal, E. (2004), *Propuesta de un modelo de conocimiento para el uso de cuestionarios en la evaluación de la usabilidad*. Tesis Doctoral, Universidad de Alcalá, pp. 1-30.
- Gediga, G., Hamborg, K. C. & Dünisch, I. (1999), 'The isometrics usability inventory: an operationalisation of ISO 9241/10', *Behavior and Information T* 18(3), 151-164.
- Guillham, B. (2000), *Developing a questionnaire*, Continuum.
- Hartson, H. R. (1996), Remote evaluation: The network as an extension of the usability laboratory, *in* 'ACM Human-Computer Interaction Conference (CHI96)', pp. 228-235.
- Hewett, T., Baecker, R., Card, S., Carey, T., Gasen, J., Manteni, M., Perlman, C., Strong, G. & Verplank, W. (1996), 'Curricula for human-computer interaction, ACM SICHI report'.
- ISIS Group (2002), 'WhizQuest: Questionnaires on the Web. <http://whizquest.isis.vt.edu/>'.
- ISO 9241 (1998), 'ISO 9241: Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability'.
- Kahn, M. & Prail, A. (1994), *Usability Inspection Methods*, John Wiley and Sons, chapter Formal usability inspections, pp. 141-172.
- Kirakowski, J. (1996), *Usability Evaluation in Industry*, Taylor and Francis, chapter The Software Usability Measurement Inventory, Background and Usage, pp. 169-179.
- Kirakowski, J. & Cierlik, B. (1998), *Transferring Usability Engineering to Industry*, TU Gdansk, chapter Usability from the top and from the bottom.
- Kirakowski, J. & Cierlik, B. (1998), Measuring the usability of Web sites, *in*



- ` Human Factors and Ergonomics Society Annual Conference (HFES98)'.
- Lewis, J. R. (1995), `IBM computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use', *International Journal of Human-Computer Interaction* 7(1), 57-78.
- Mack, R. L. & Burdett, J. M. (1992), *The Psychology of Expertise: Cognitive Research and Empirical AI*, Springer-Verlag, chapter When novice elicit knowledge: Questionasking in designing, evaluating and learning to use software, pp. 245-268.
- Mack, R. L. & Nielsen, J. (1994), *Usability Inspection Methods*, John Wiley and Sons, chapter Executive summary, pp. 1-23.
- Mandel, T. (1997), *The elements of user interface design*, Wiley and Sons.
- Mayhew, D. (1999), *The usability engineering lifecycle*, Morgan Kaufmann.
- Mayhew, D. & Mantei, M. (1994), *Cost justifying usability*, Academic Press, chapter A basic framework for cost-justifying usability engineering, pp. 9-43.
- Muñiz, J. (1996), *Psicometría*, Universitas, chapter Fiabilidad.
- Nielsen, J. (1993), *Usability Engineering*, Morgan Kaufmann.
- Nielsen, J. (1994b), *Usability Inspection Methods*, John Wiley and Sons, chapter Heuristic Evaluation, pp. 25-62.
- O'Donnell, P. J., Scobie, G. & Baxter, I. (1989), *People and Computers VI*, Ellis Horwood, chapter The use of focus groups as an evaluation technique in HCI, pp. 211-224.
- O'Malley, C. E., Draper, S. W. & Riley, M. S. (1984), *Constructive interaction: A method for studying human-computer-human interaction*, in 'Intl. Conference on Human-Computer Interaction (INTERACT84)', pp. 269-274.
- Oppenheim, A. (1992), *Questionnaire design, interviewing and attitude measuring*, Printed Pub.
- Paz, M. D. (1996), *Psicometría*, Universitas, chapter Validez.
- Perlman, G. (2002), 'Web-based user interface evaluation with questionnaires. <http://www.acm.org/perlman/question.html>'.
- Root, R. W. & Draper, S. (1983), *Questionnaires as a software evaluation tool*, in `ACM Conference Human Factors in Computing Systems (CHI83)', pp. 83-87.
- Rosson, M.B., Carroll, J.M., Seals, C., & Lewis, T. 2002. *Community design*



of community simulations. *Proceedings of Designing Interactive Systems: DIS 2002*. New York: ACM.

Vora, P. & Helander, M. (1995), *Symbiosis of Human and Artifacts*, North-Holland_ chapter A Teaching method as an alternative to the concurrent think-aloud method for usability testing, pp. 375-380.

Wharton, C., Rieman, J., Lewis, C. & Polson, P. (1994), *Usability Inspection Methods*, John Wiley and Sons, chapter The cognitive walkthrough method: A practitioner's guide, pp. 105-140.

Whitefield, A., Wilson, F. & Dowell, J. (1991), 'A framework for human factors evaluation', *Behaviour and Information Technology* 10(1), 65-79.

Wixon, D., Jones, S., Tse, L. & Casaday, G. (1994), *Usability Inspection Methods*, John Wiley and Sons, chapter Inspections and design reviews: Framework, history, and reflection, pp. 79-104.

Referencias web

[1] Document Object Model (DOM),
<http://www.librosweb.es/javascript/capitulo5>

[2] Eclipse, <http://www.eclipse.org>

[3] Hibernate, <https://www.hibernate.org/>

[4] Interacción Persona-Ordenador (IPO), <http://www.aipo.es/>

[5] Java, <http://java.sun.com/>

[6] JBoss, <http://labs.jboss.com>

[7] JUnit, www.junit.org

[8] MySQL, <http://www.mysql.com>

[9] Struts, <http://struts.apache.org>